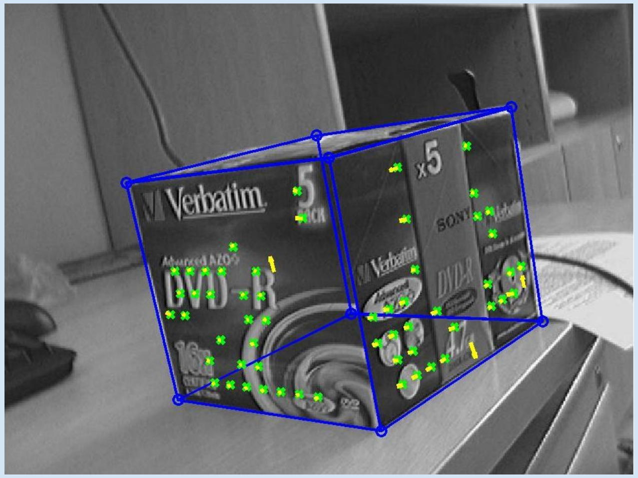
An Optimal Sequence of Learned Motion Estimators

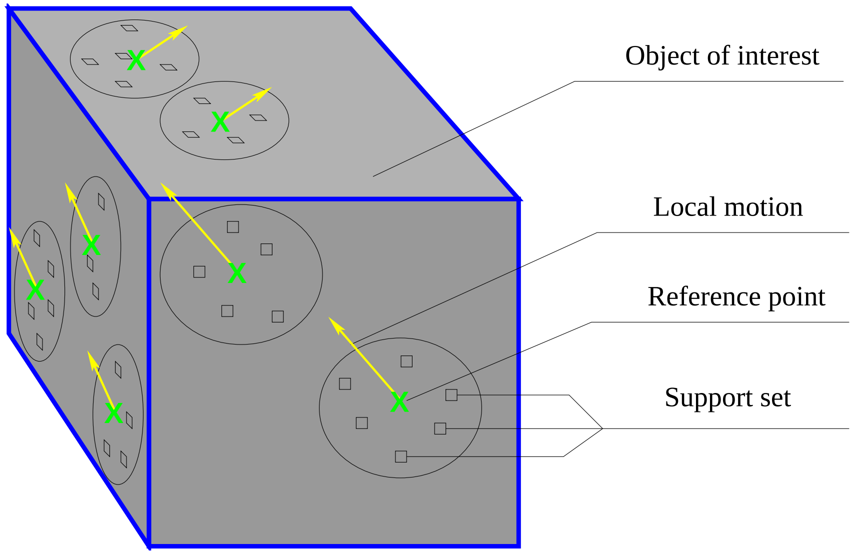
Karel Zimmermann[1], Jiří Matas[1],
Tomáš Svoboda[1,2]

[1]: Center for Machine Perception
[2]: Center for Applied Cybernetics
Czech Technical University
Prague, Czech Republic

Introduction



Video: 3D tracking

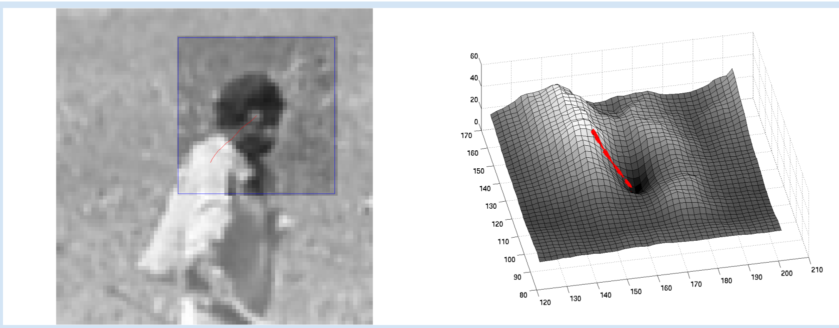


Tracking objectives:

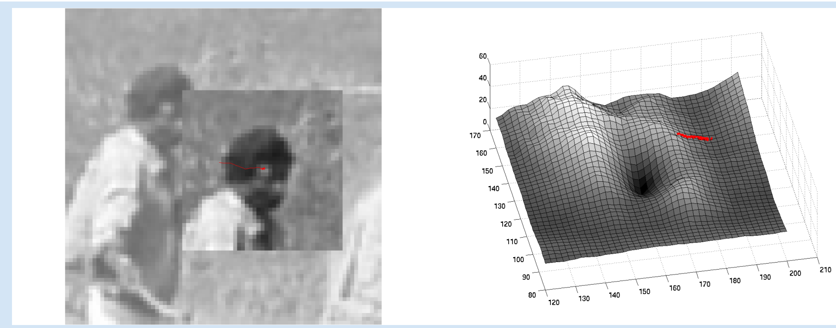
- Fast
- Accurate
- Robust

State-of-the-art: Tracking by gradient optimization

- **Minimize dissimilarity:** $\mathbf{t} = \arg\min_{\mathbf{t}} \sum \left(I(\mathbf{x}+\mathbf{t}) - J(\mathbf{x})\right)^2$
  - [1] S.Baker and I.Matthews, Lucas-Kanade 20 Years On: A Unifying Framework, International Journal of Computer Vision, pp.221-255, 2004



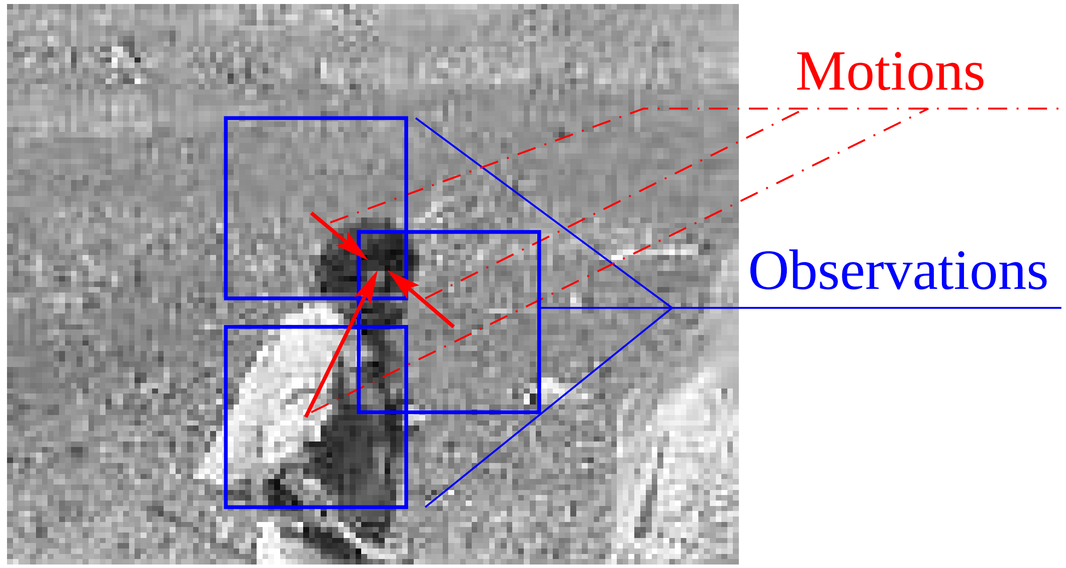
Video: example of KLT convergence

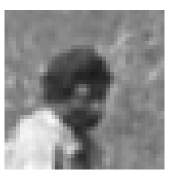


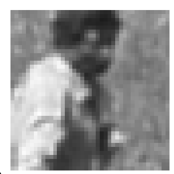
Video: example of KLT divergence

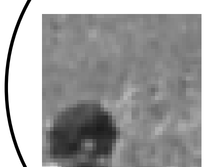
- Drawbacks:
  - Convergence to a local minimum
  - Unknown basin of attraction
  - Criterial function

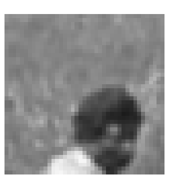
State-of-the-art: Tracking by regression



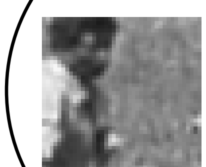
Φ () $= (0,0)^T$

Φ () $= (-14,2)^T$

Φ () $= (14,-14)^T$

Φ () $= (12,7)^T$

Φ () $= (-9,18)^T$

Φ () $= (-16,-12)^T$

- There is an inverse relation approximated by mapping

$$\Phi : \text{intensities around a point} \rightarrow \text{motion}$$

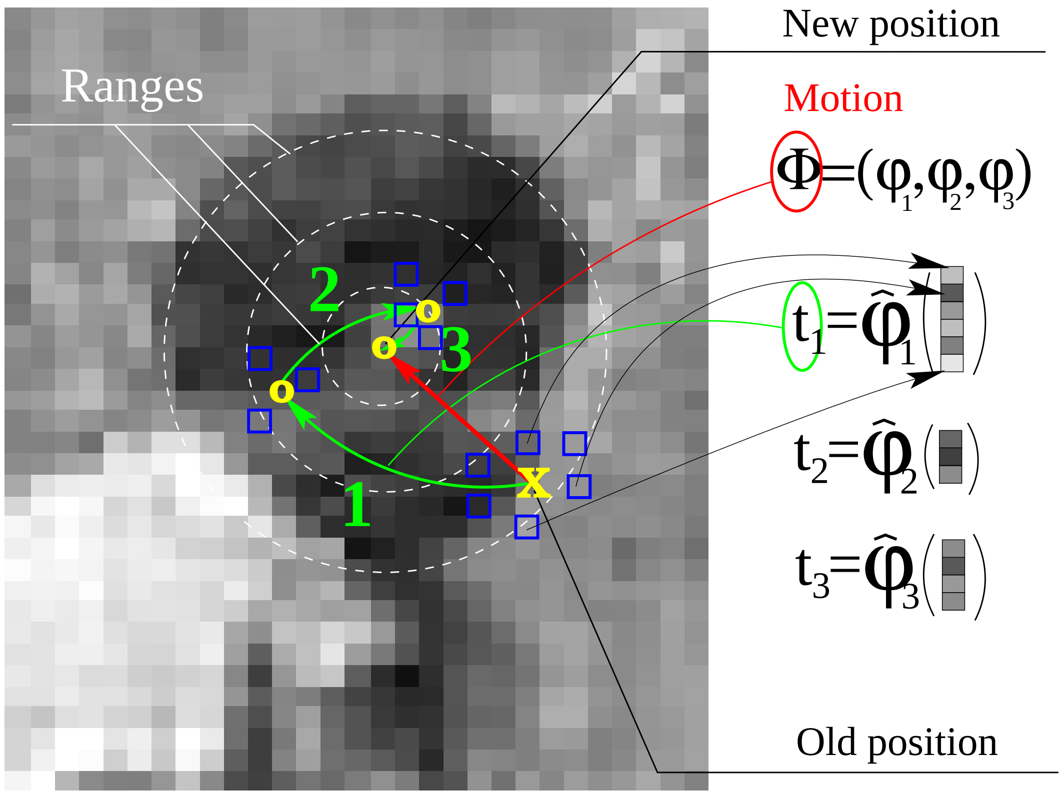
State-of-the-art: Tracking by regression

- **Linear motion regression:** $\mathbf{t} = \mathtt{H}\big(I(\mathbf{x}) - J(\mathbf{x})\big)$
  - [2] T.Cootes, G.Edwards, and C.Taylor, Active Appearance Model, Pattern Analysis and Machine Intelligence, pp.681-685, 2001
  - [3] F.Jurie and M.Dhome, Real time robust template matching, British Machine Vision Conference, pp.123-131, 2002

- **Non-linear motion regression:** RVM
  - [4] O.Williams, A.Blake and R.Cipolla, Sparse Bayesian Learning for Efficient Visual Tracking, Pattern Analysis and Machine Intelligence, pp.1292-1304, 2005

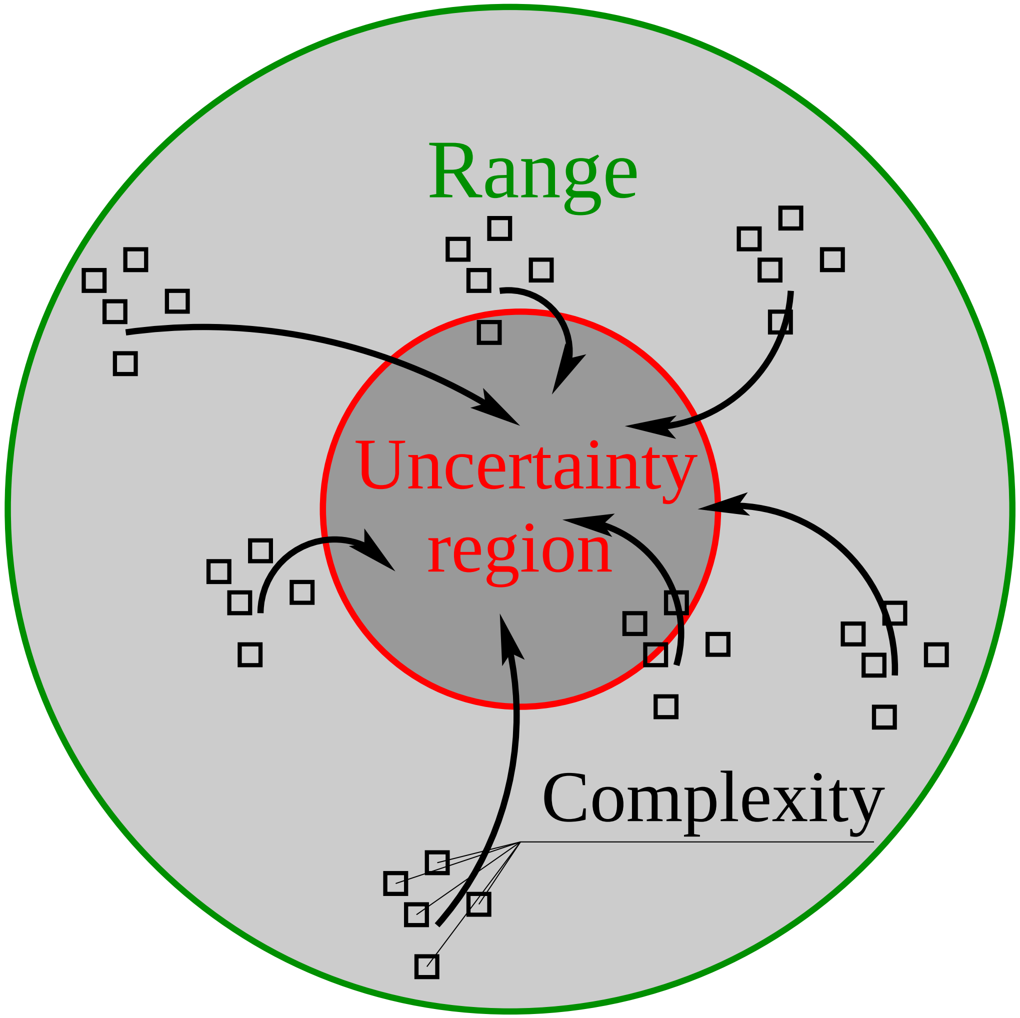
Our approach

- **Sequential motion regression:** $\mathbf{t} = \varphi_h\Big(\ldots I\big(\mathbf{x} + \varphi_1(I(\mathbf{x}))\big)\Big)$



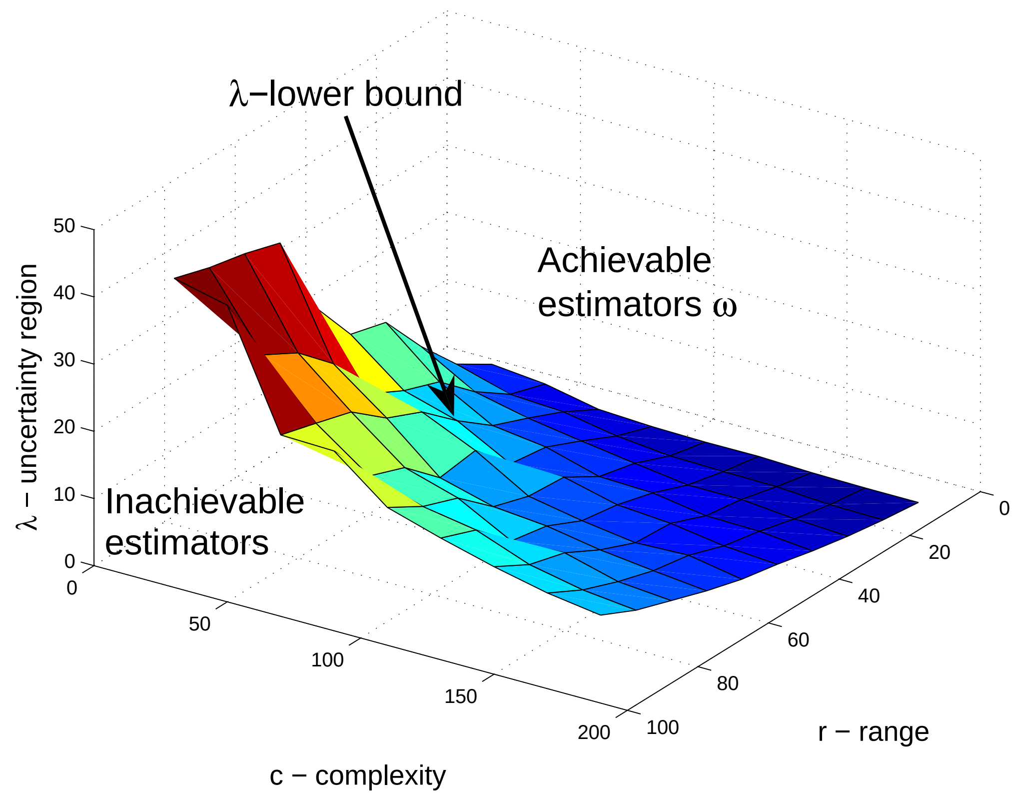
- We are looking for a sequence of predictors $\Phi = [\varphi_1, \varphi_2, \ldots \varphi_h]$ with the lowest complexity.
  - How many iterations h are required?
  - How many pixels are neccesary for each iteration?
  - What neighbouring pixels are used?

Uncertainty region



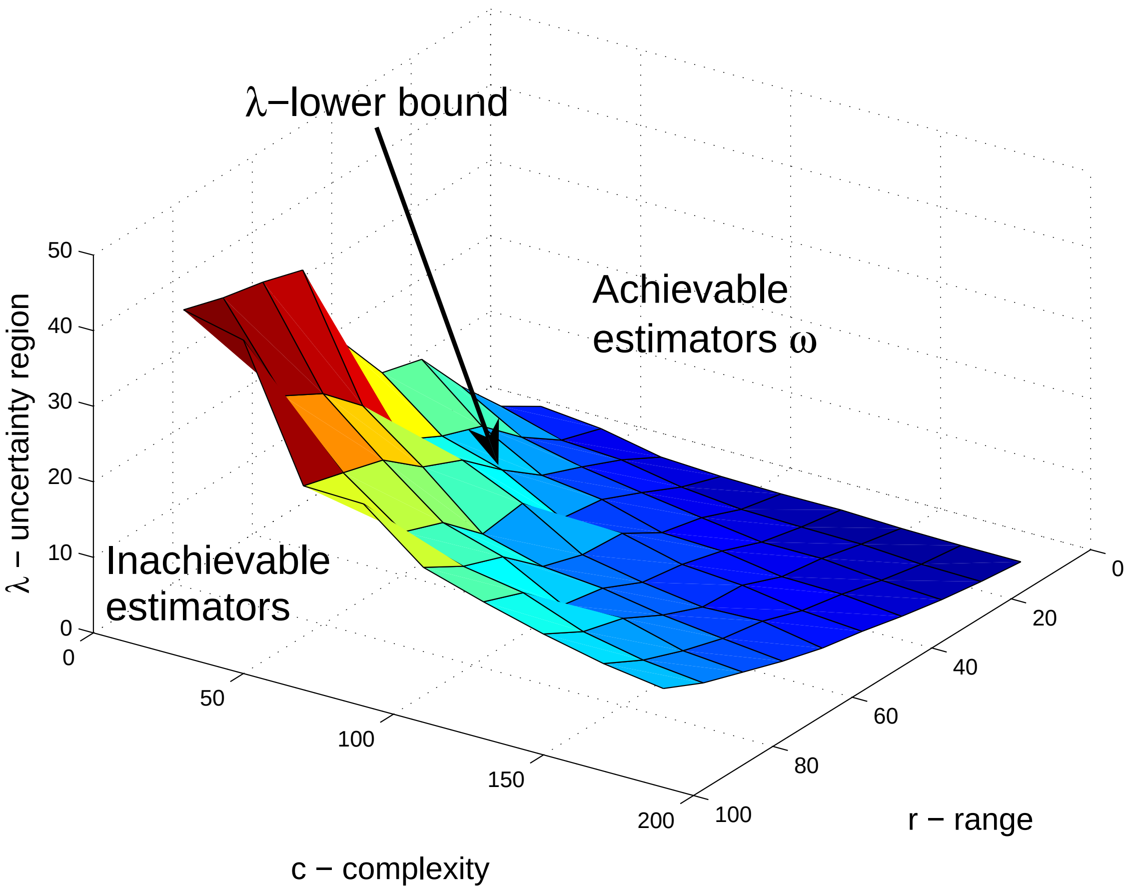
- **Range** r the set of admissible motions.
- **Complexity** c cardinality of support set.
- **Uncertainty region** λ the region within which all the estimations lie.

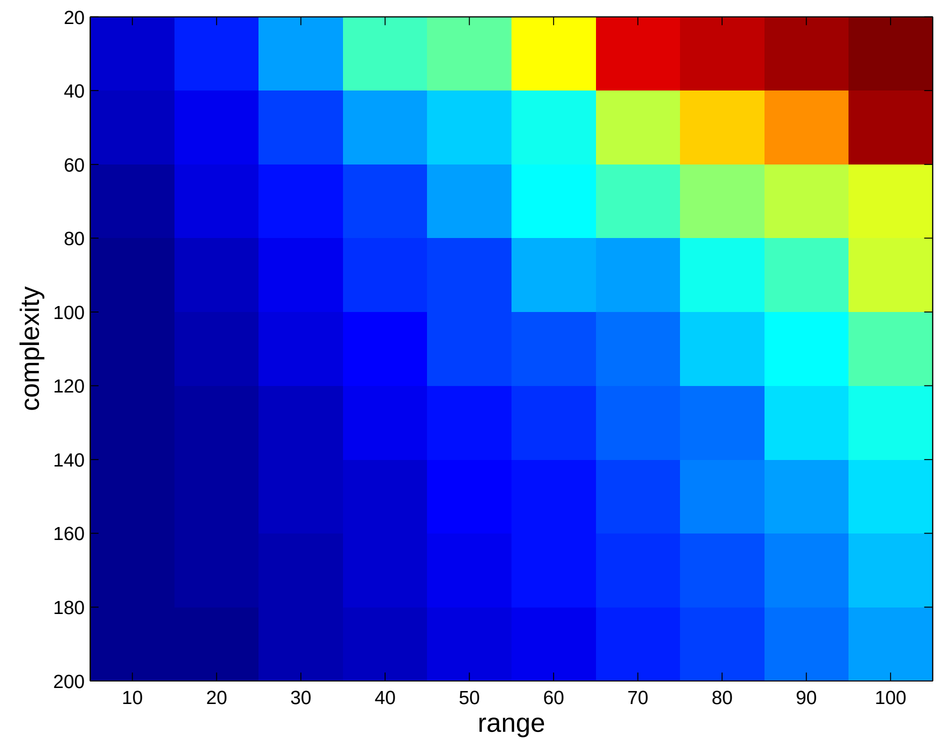
Optimal sequence of optimal predictors



- **Predictors** $\phi_i(c, r, \lambda)$ lie in a subspace of the (c, r, λ)-space.
- **Optimal sequence of predictors** is a sequence $\Phi = [\varphi_1, \varphi_2, \ldots \varphi_h]$ with the lowest total complexity $\sum c_i$ given:
  - range r_1 of the first predictor
  - uncertainty region λ_h of the last predictor.
  - $r_{i+1} \geq \lambda_i,\ i = 1, \ldots, h-1$.

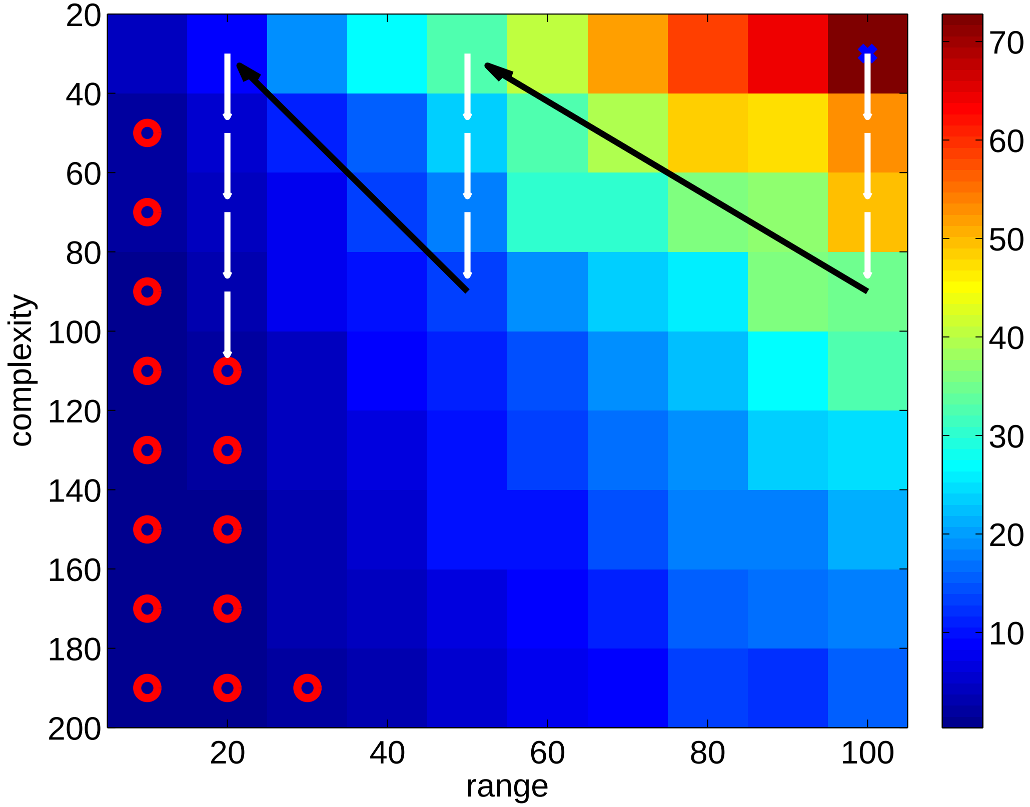
An optimal sequence

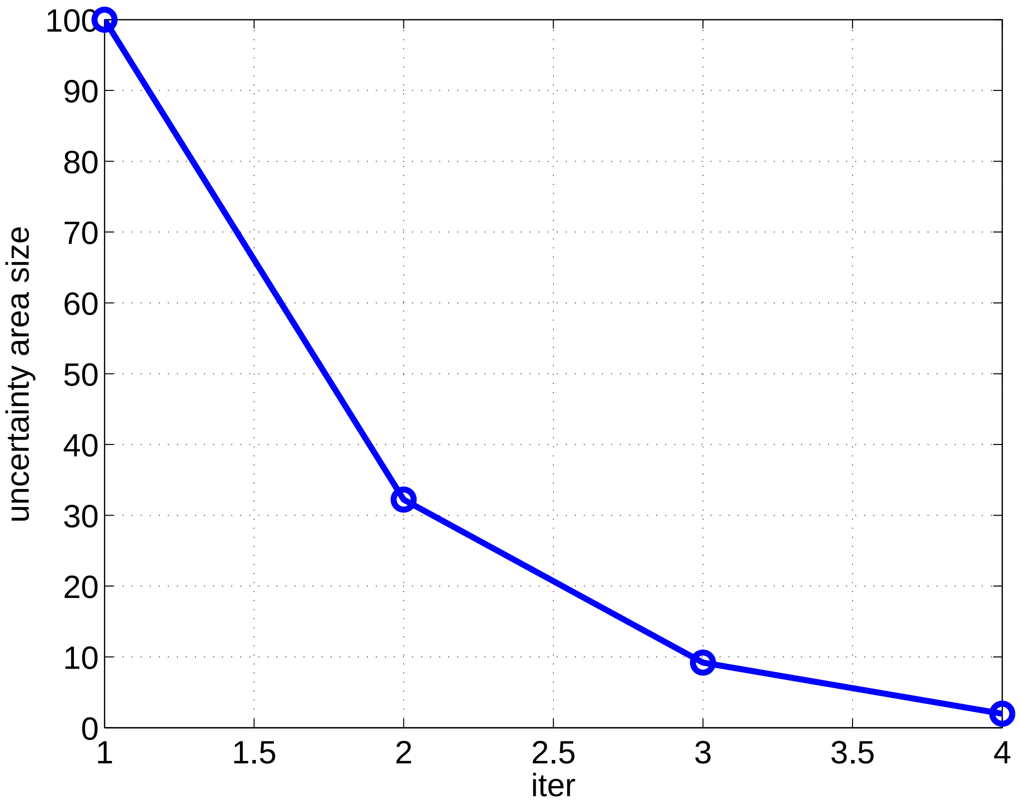




- Only those predictors lying on the λ-lower bound of the set of achievable predictors can create an optimal sequence $\hat{\Theta}$.
- Given (c,r), minimax task is solved to find the predictor with the smallest uncertainty region.
- Color codes the size of the uncertainty region.

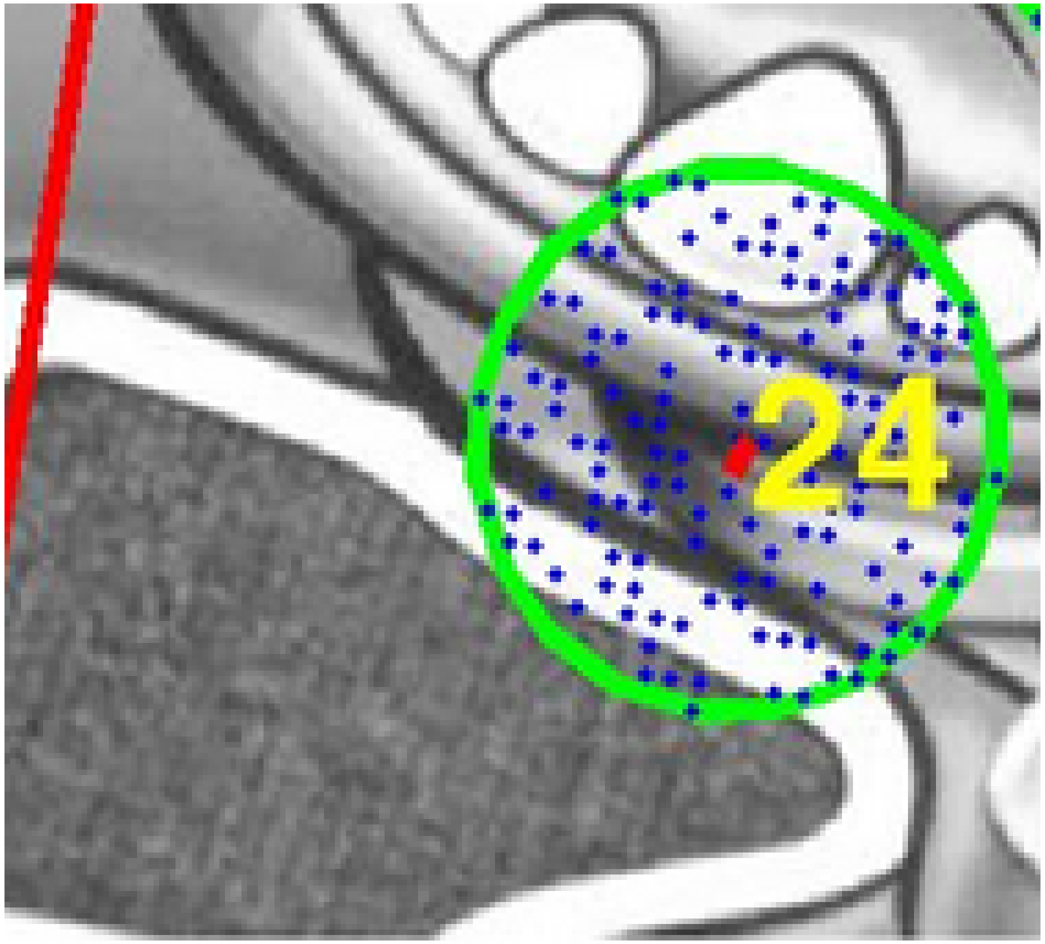
Searching for an optimal sequence.

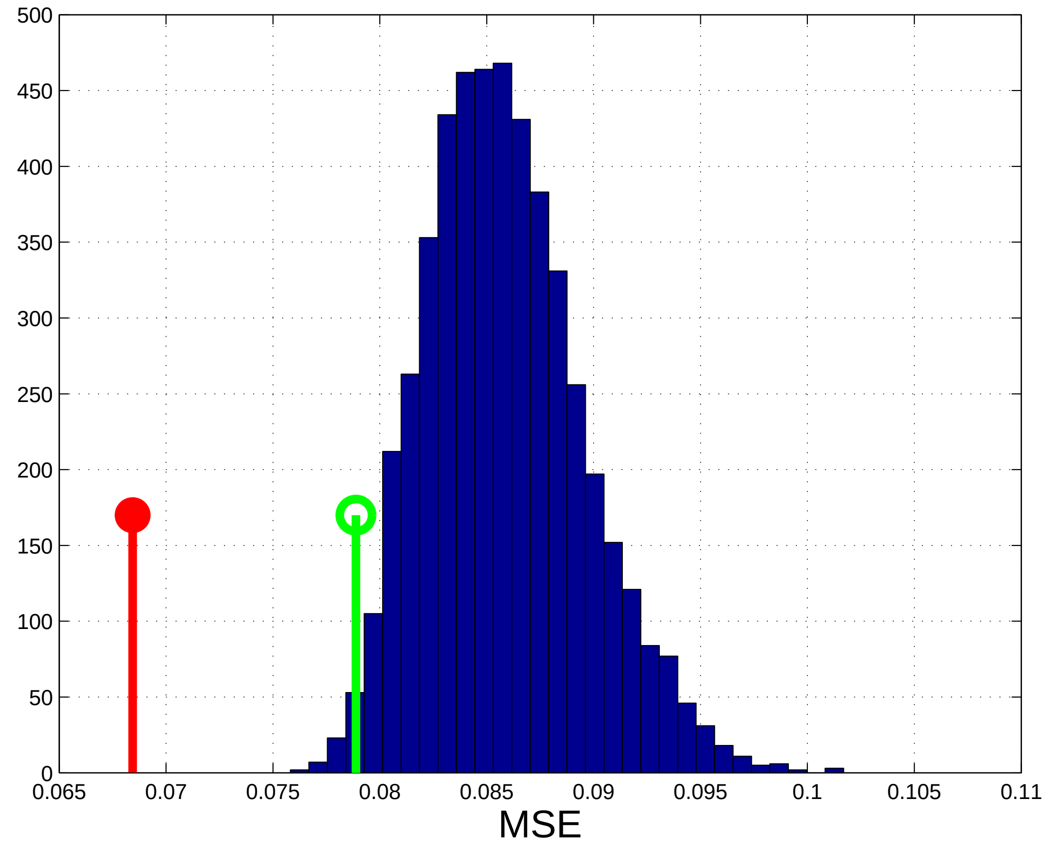




- Dynamic programming searches for an optimal sequence of predictors.
- The algorithm searches for the cheapest path to a sufficiently small uncertainty region.
- In each state either complexity is increased or the next iteration initialized.

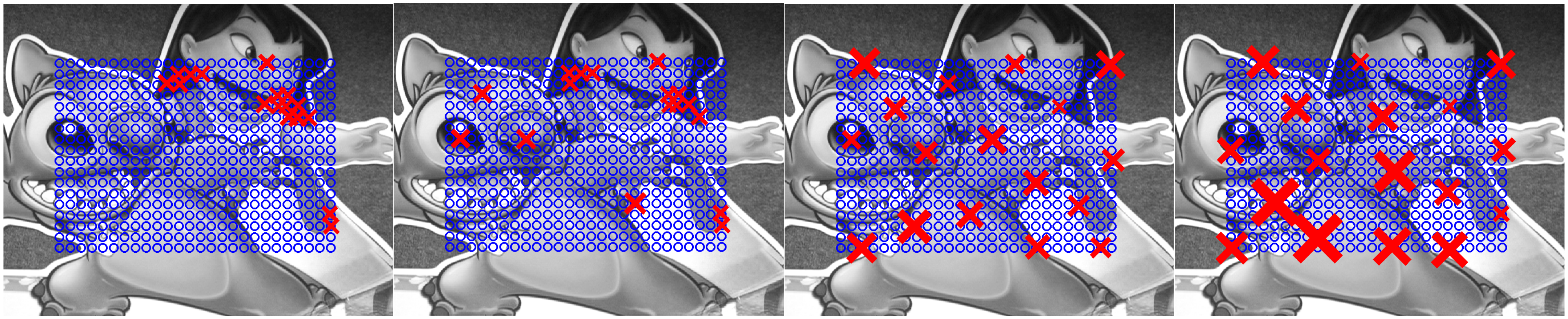
Support set selection



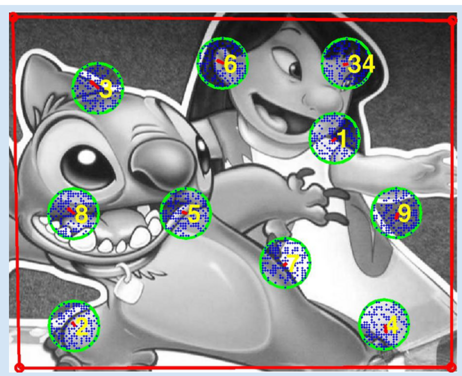


- Greedy LSQ selection (red) of an efficient support set.
- Much better than 1%-quantile (green) achieavable by randomized sampling

Online selection of an active predictor set



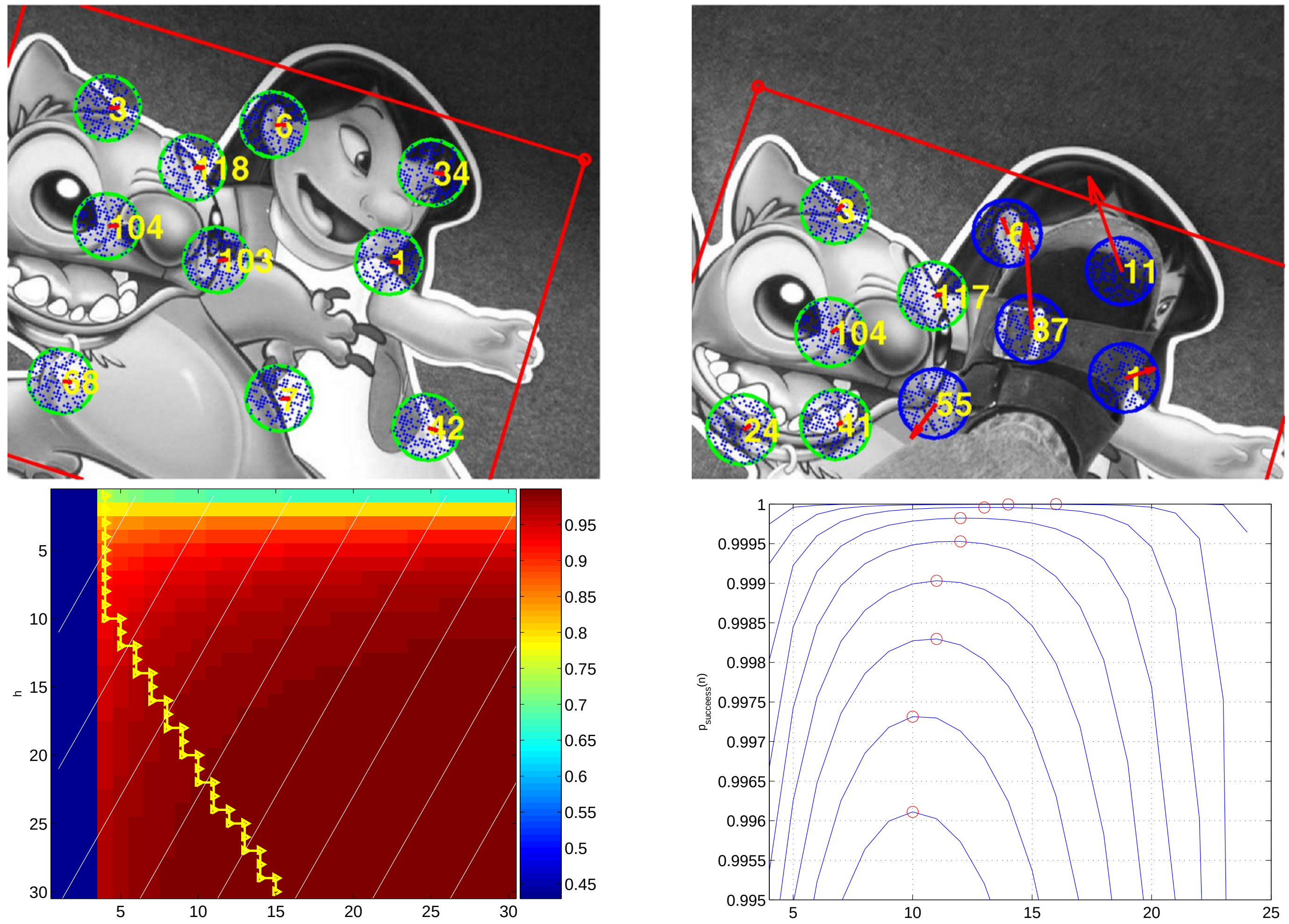
a) w=0 b) w=0.1 c) w=0.5 d) w=1



Video: occlusions

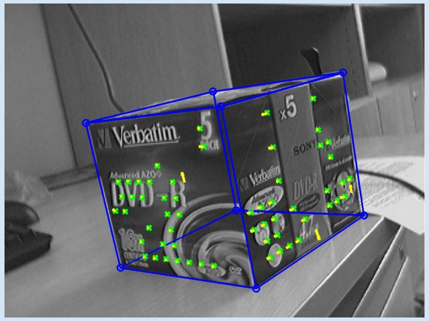
- Greedy online selection.
- Trade-off between abilities of local predictors and coverage of an object.
- Strong features may not provide good tracking results.

RANSAC iterations × Number of predictors



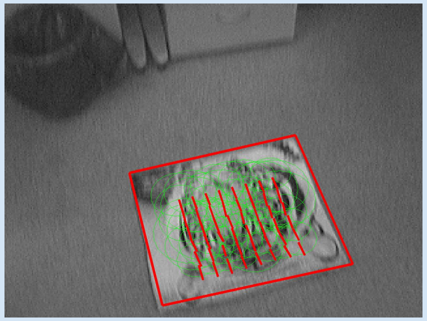
- Probability of successful tracking as a function of number of ransac iterations and predictors.
- We maximize the probability, given a time, we are allowed to spent with the motion estimation in the actual frame,

Motion blur, fast motion, views from acute angles and other image distortions.

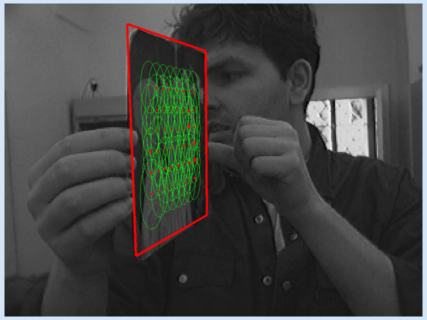


Video: 3D tracking

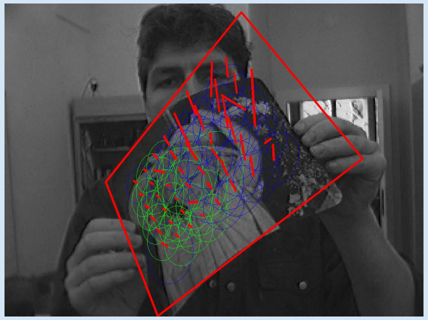
Video: fast motion



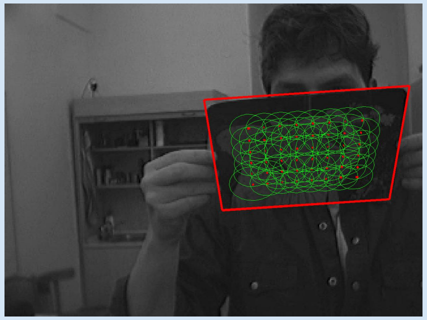
Video: blured motion



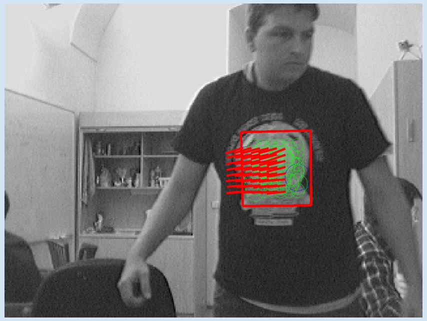
Video: acute angles



Video: bending

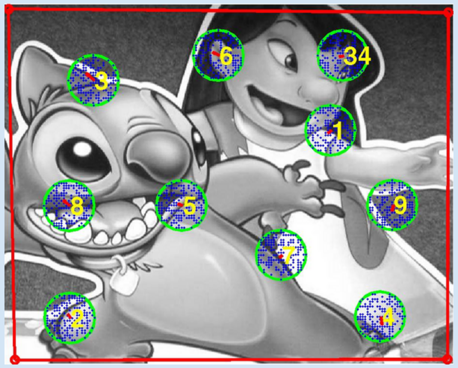


Video: illumination



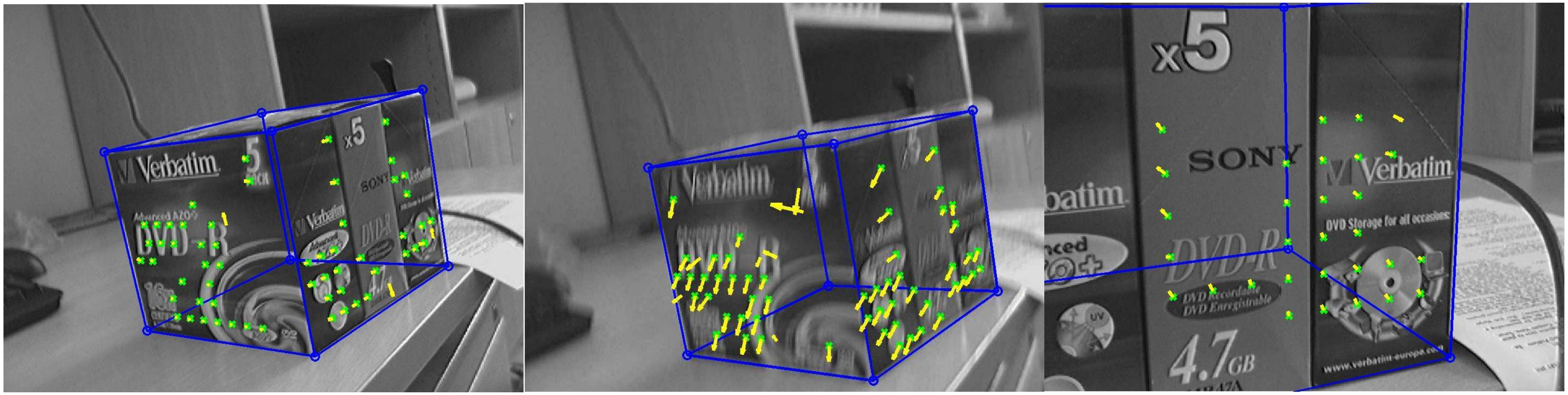
Video: pseudo planar

Video: occlusions



Video: occlusions

Experiments: 3D fast blured tracking

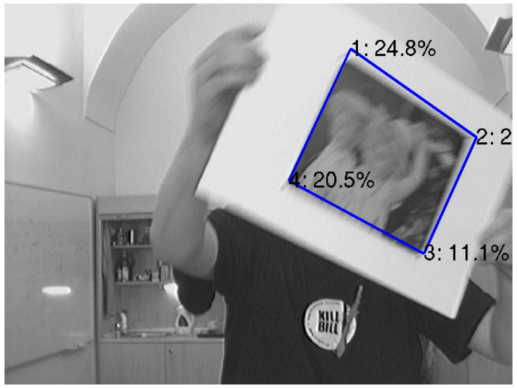


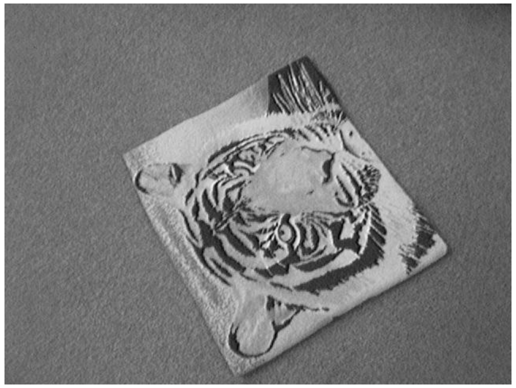
a) slow motion b) fast blured motion c) close view

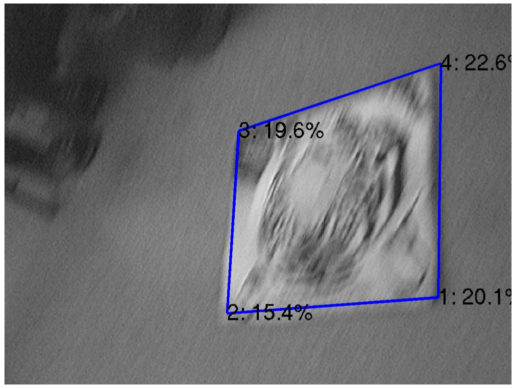
Experiments: Results on sequences 2000-7000 frames.

object	processing	loss-of-locks	mean-error
mouse pad minmax	18.9fps	13/6935	[1.3%, 1.8%, 1.5%, 1.6%]
mouse pad sift	0.5fps	281/6935	[1.6%, 1.2%, 1.5%, 1.4%]
towel minmax	21.8fps	5/3229	[3.0%, 2.2%, 1.4%, 1.9%]
phone minmax	16.8fps	20/1799	[1.2%, 1.8%, 2.6%, 1.9%]

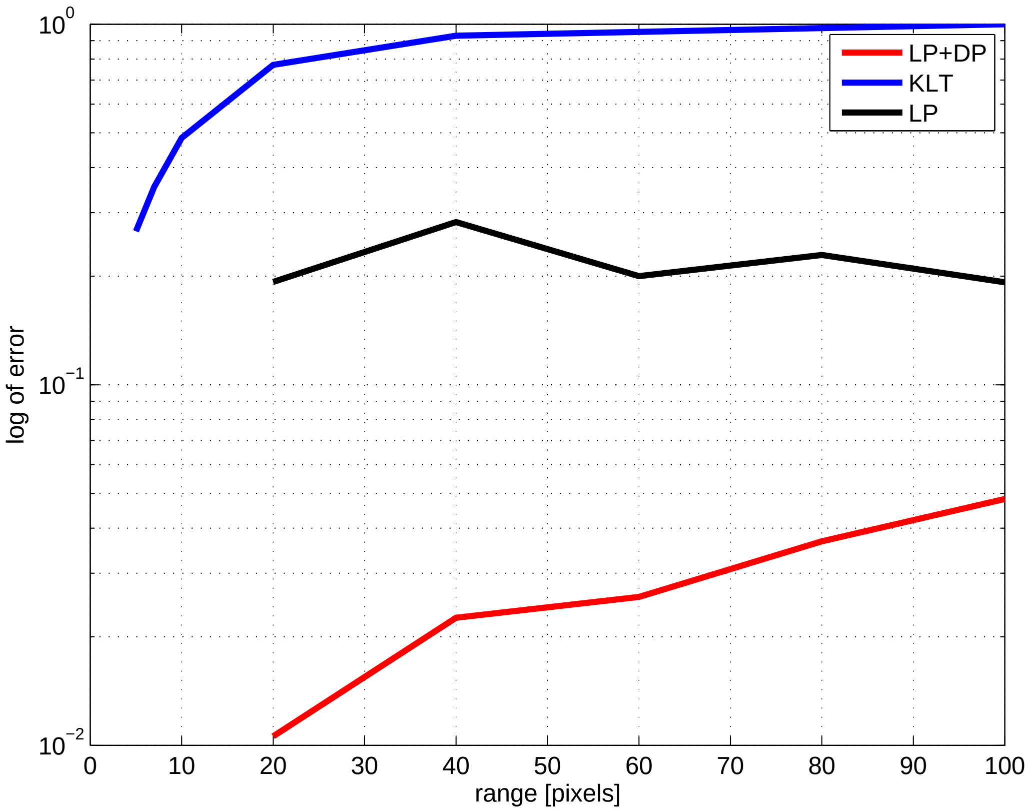
- Data captured at 22.7fps frame-rate.
- Comparison to SIFT detector.

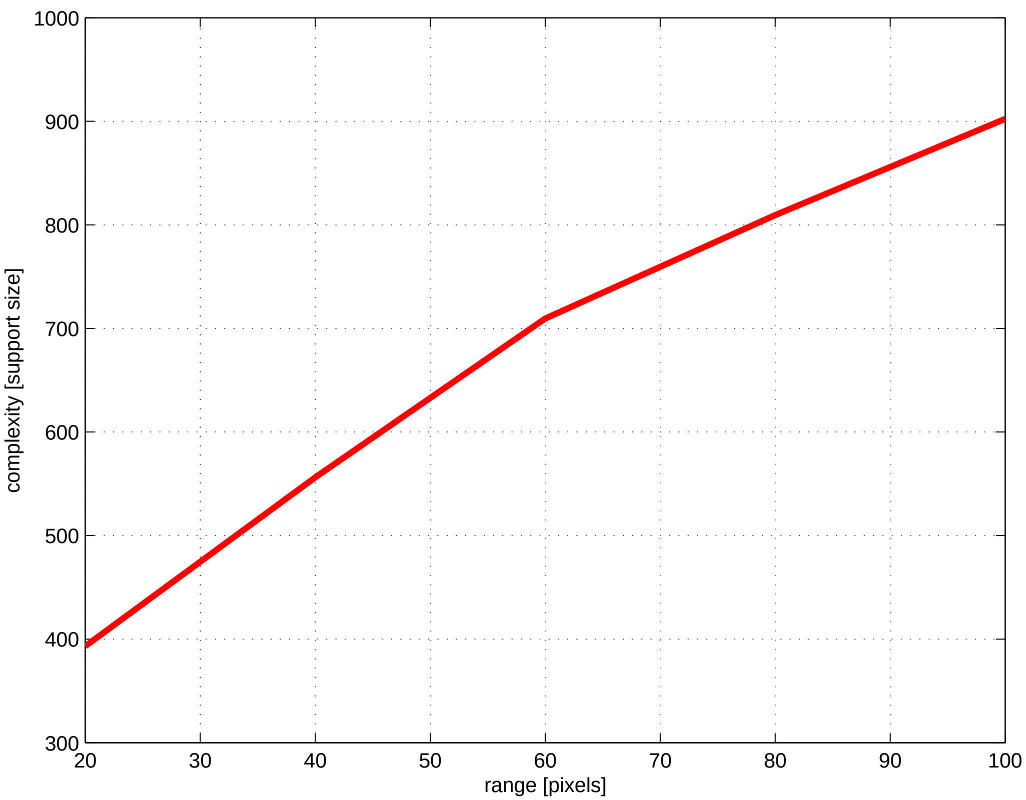






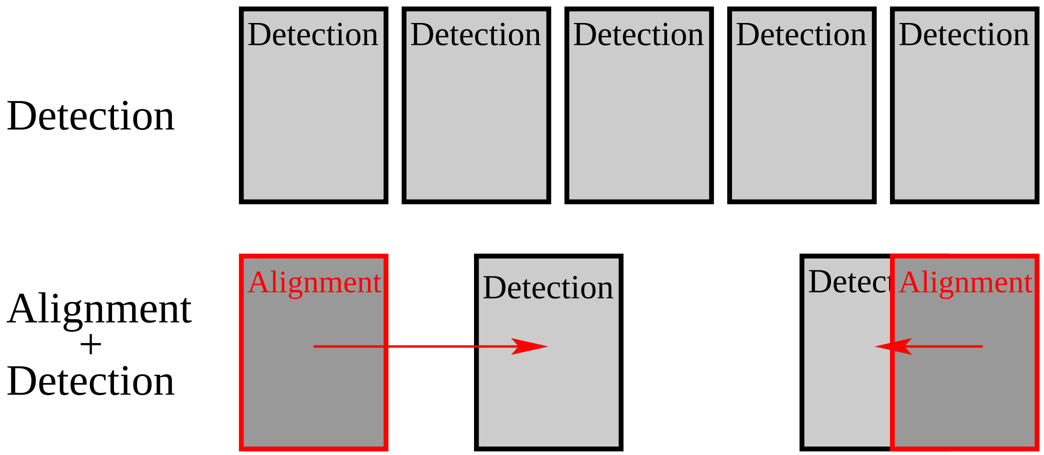
Experiments: Comparison with KLT.

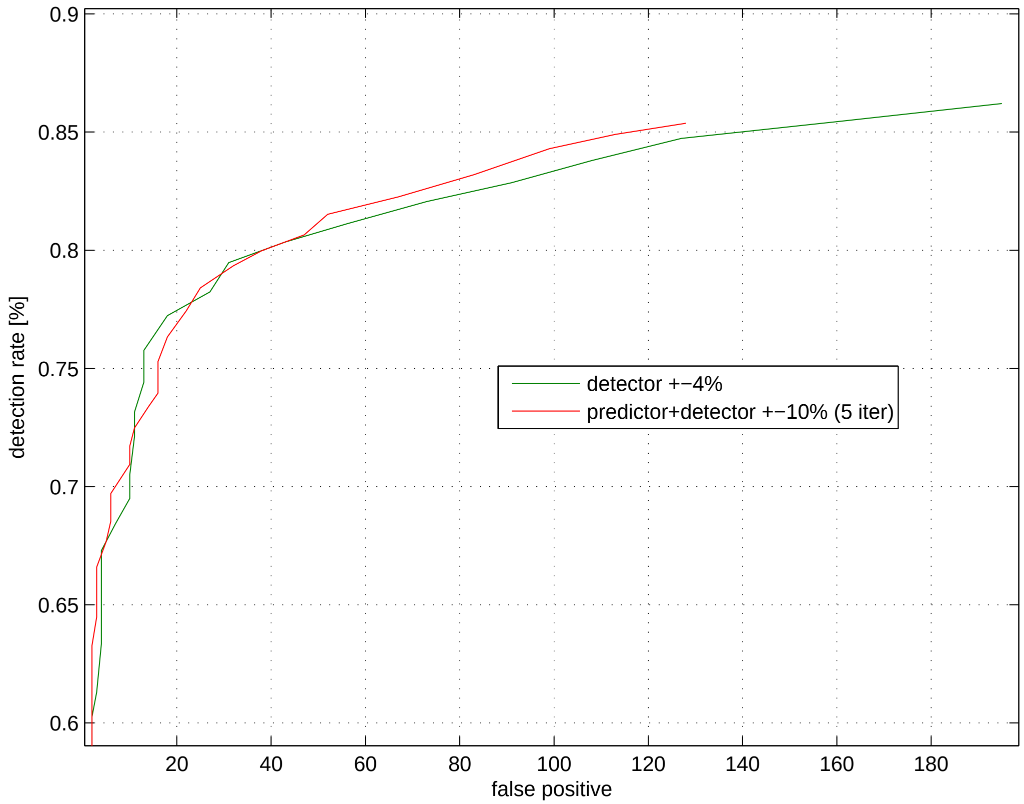




- Much lower complexity and substantionally smaller error rate.
- If the number of iteration is constant than error rate is independent of the range.

Experiments: Application to a face detector.



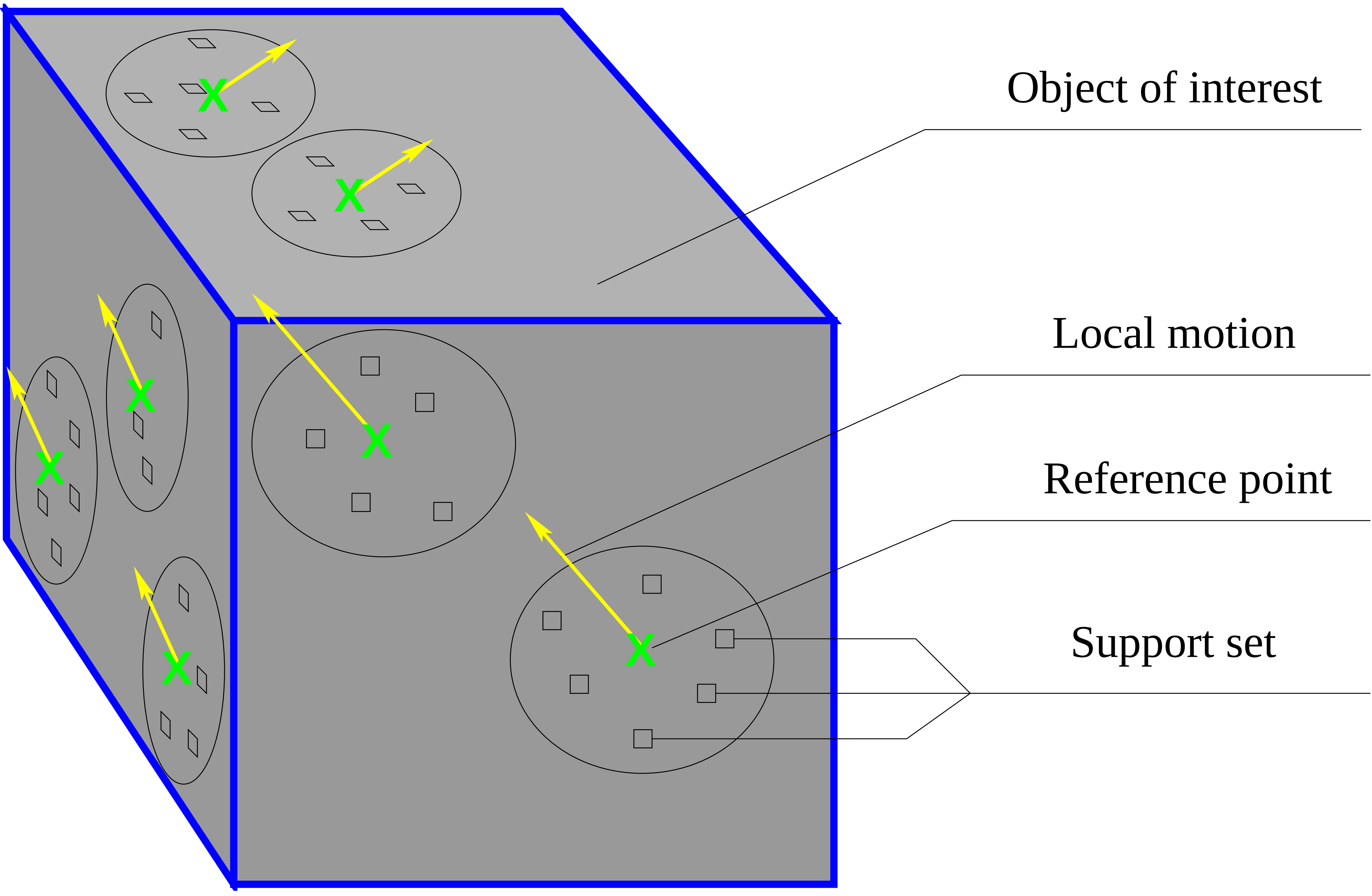


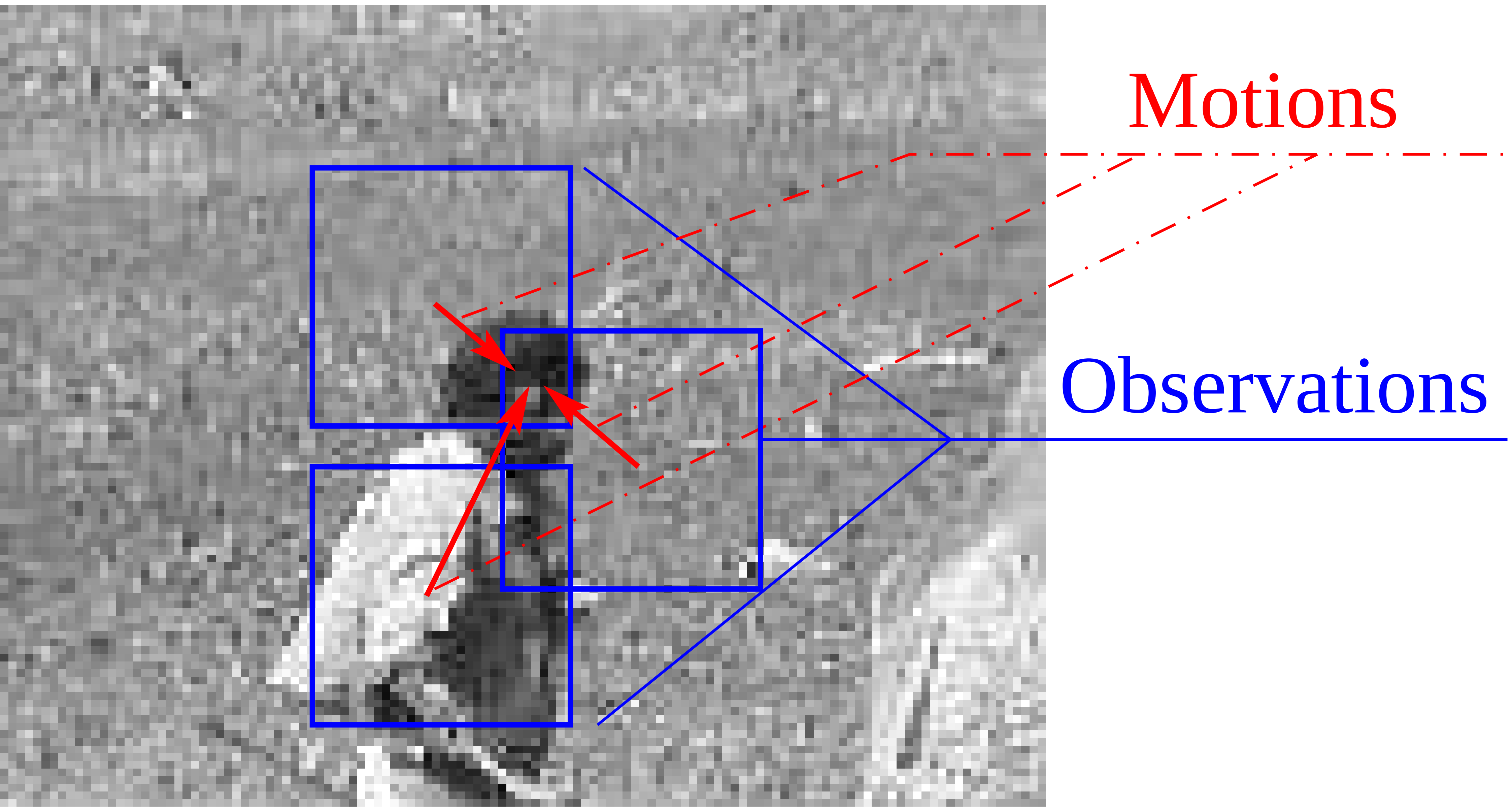
	memory accesses	summations	multiplications
Alignment	15	30	30
Detector	25	25	0
Align+Det	6.5	9	5

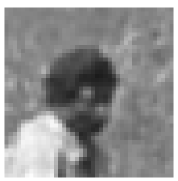
Conclusion

- Drawbacks:
  - Learning required.
  - Predictor range is limited by the size of the object.
- Advantages:
  - Very fast motion estimation ($30\mu s$ per predictor).
  - Ability to cover arbitrary cases (bluring, change of appearance).
  - Automatic setup of tracking procedure.

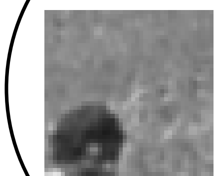
m p

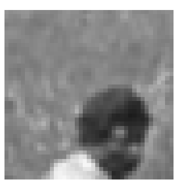

Object of interest
Local motion
Reference point
Support set


Motions
Observations

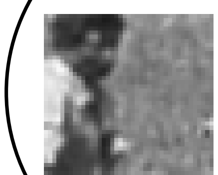
$\Phi\Bigg($  $\Bigg) = (0,0)^T$

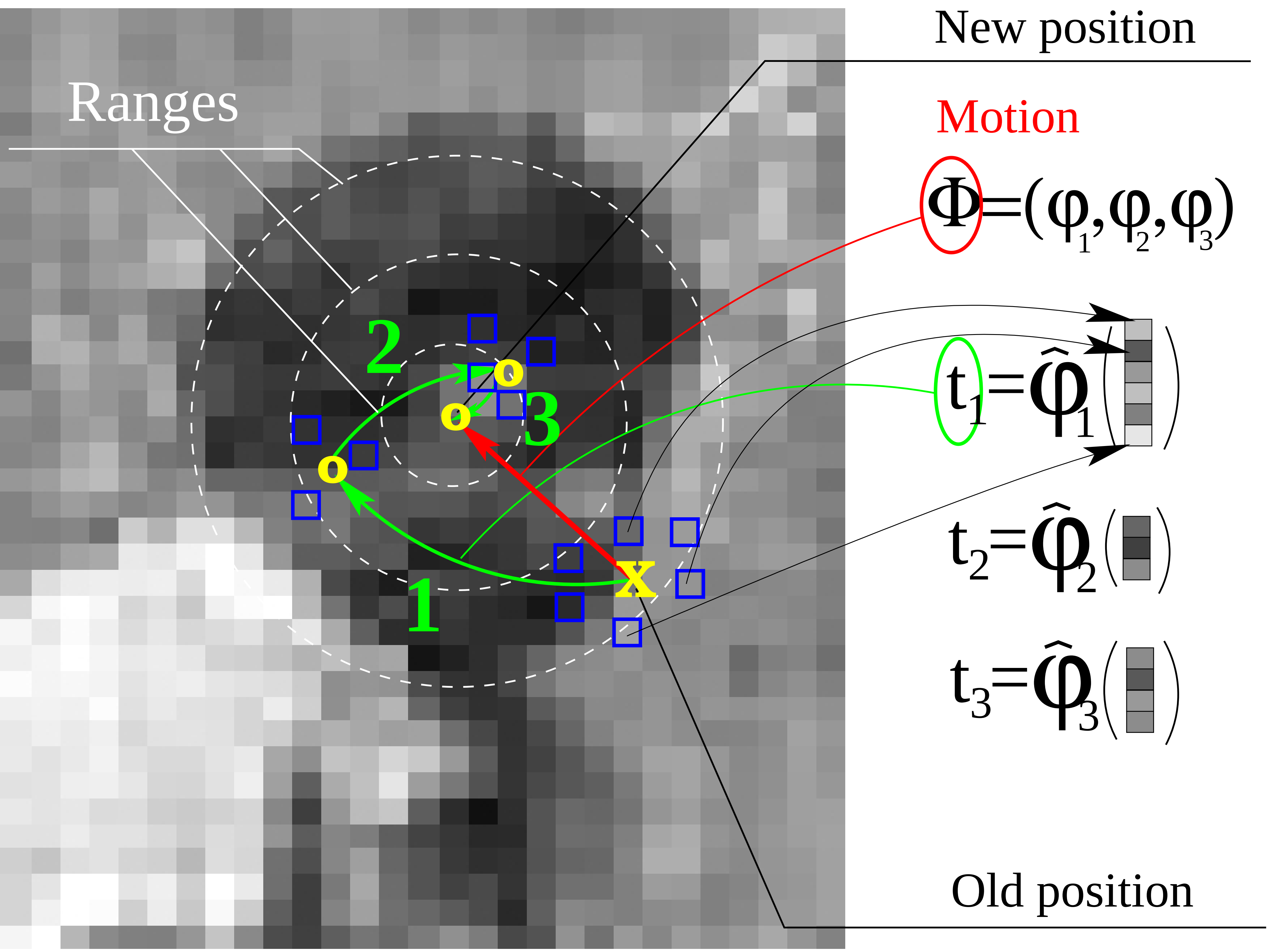
$\Phi\Bigg($  $\Bigg) = (-14,2)^T$

$\Phi\Bigg($  $\Bigg) = (14,-14)^T$

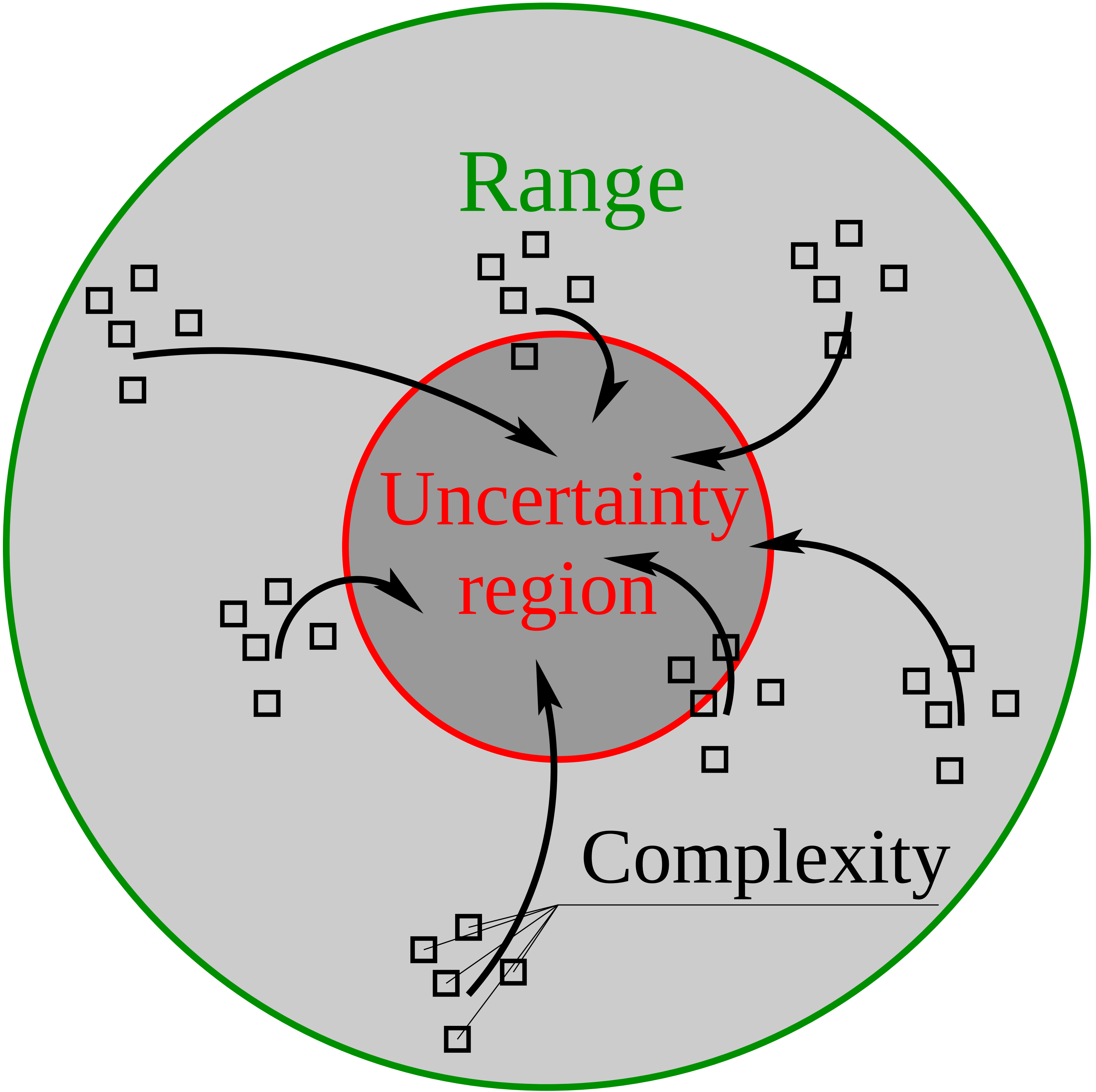
$\Phi\Bigg($  $\Bigg) = (12,7)^T$

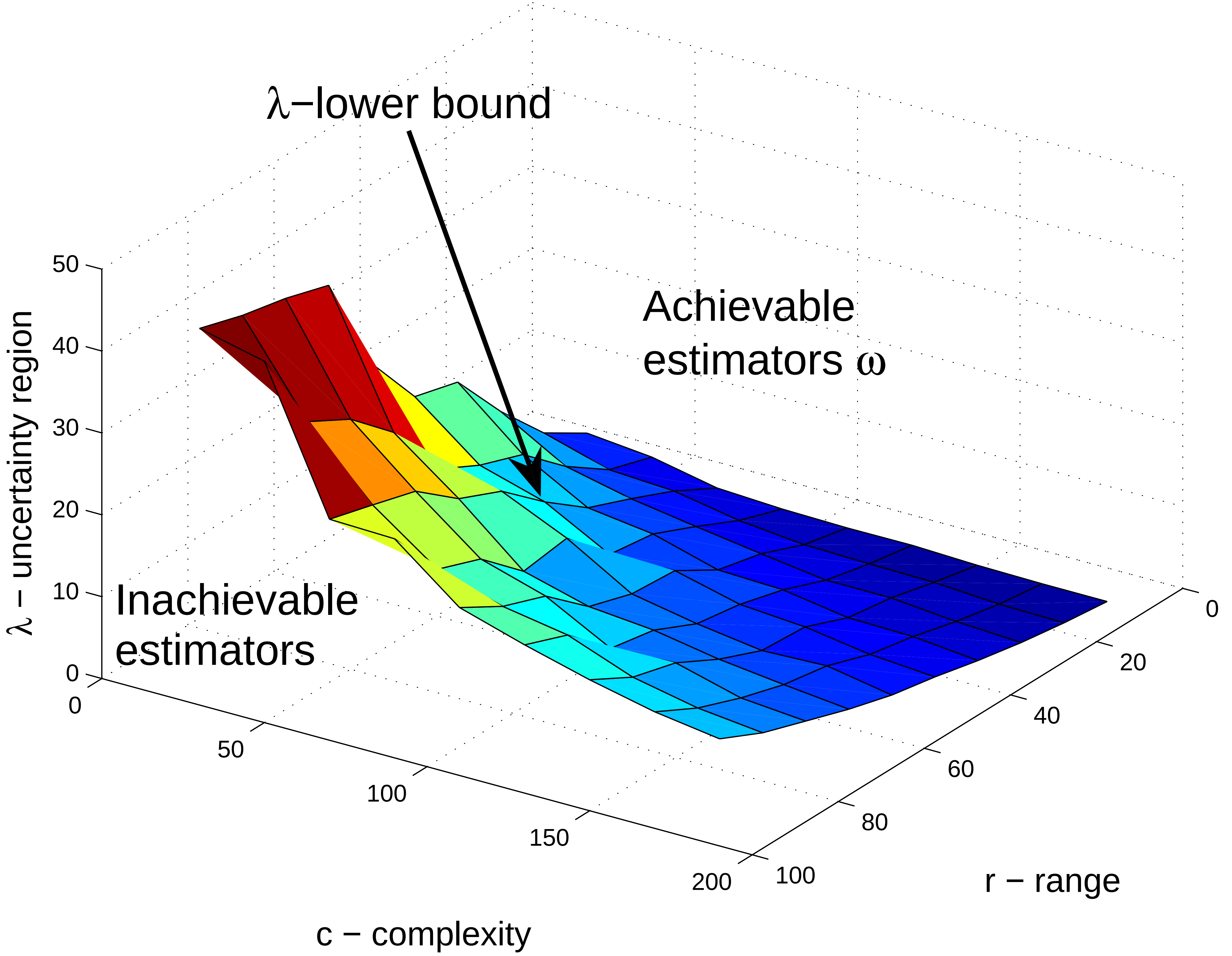
$\Phi\Bigg($  $\Bigg) = (-9,18)^T$

$\Phi\Bigg($  $\Bigg) = (-16,-12)^T$

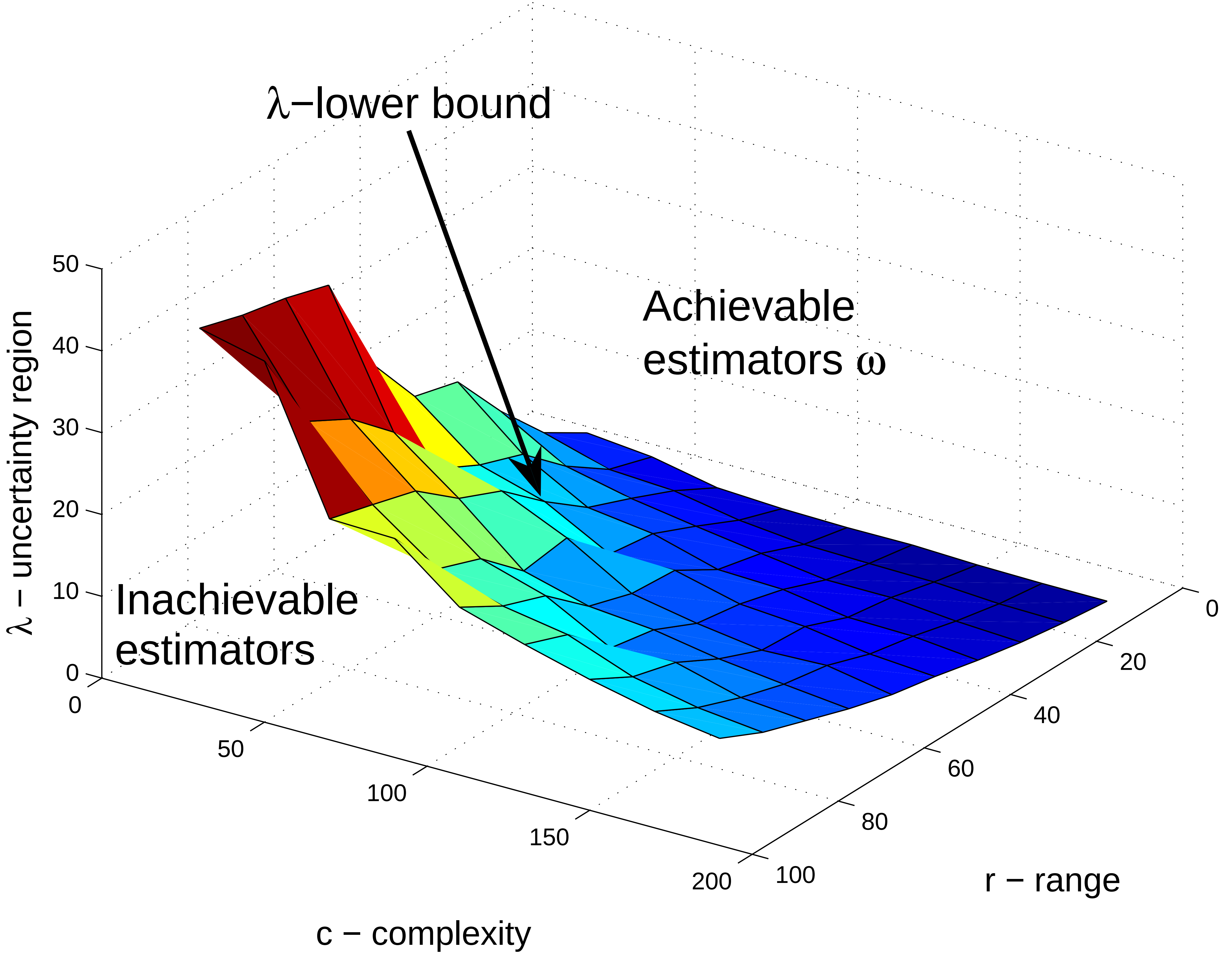


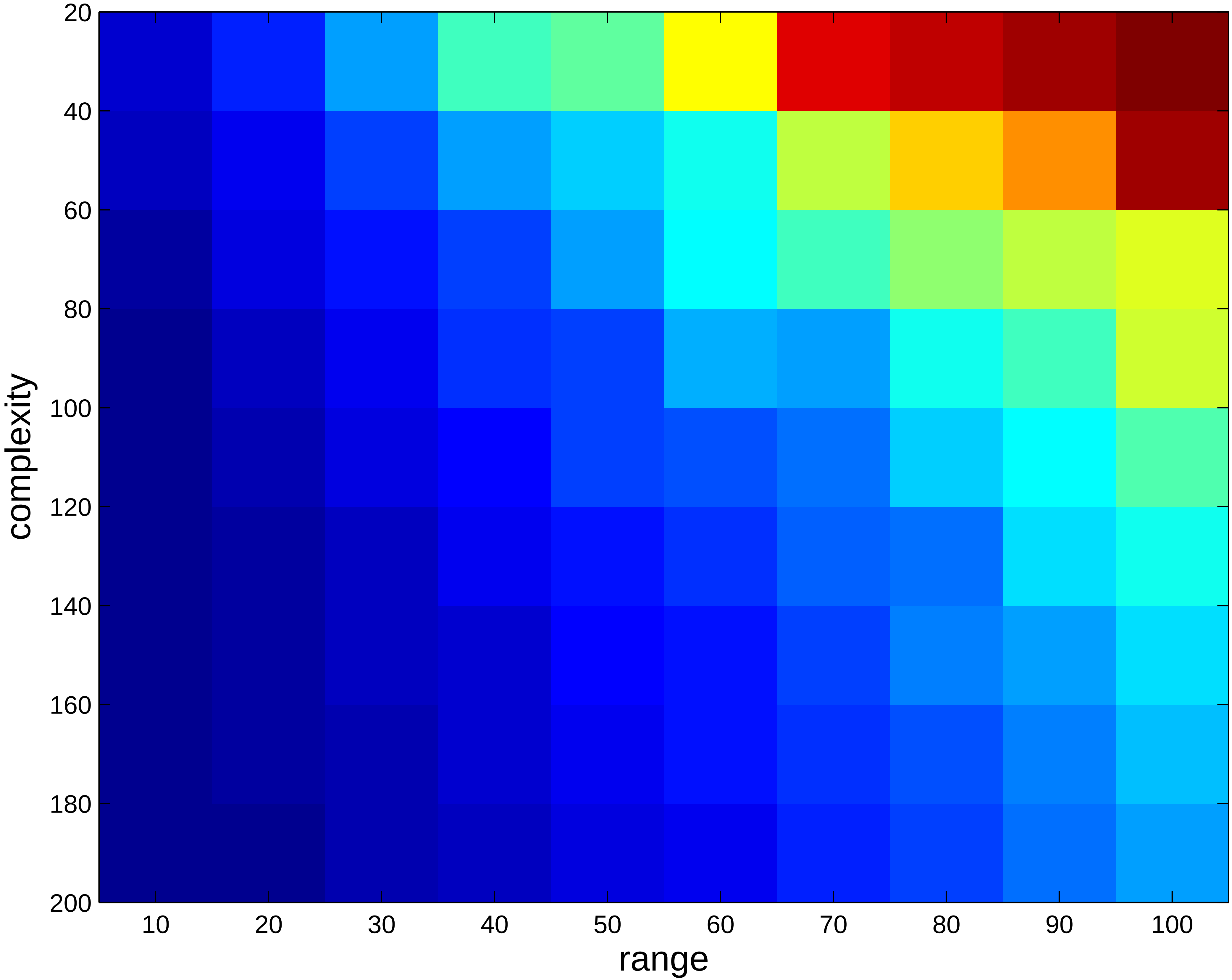
New position
Ranges
Motion
$\Phi = (\varphi_1, \varphi_2, \varphi_3)$
2
o
o
3
o
1
x
$t_1 = \hat{\varphi}_1$
$t_2 = \hat{\varphi}_2$
$t_3 = \hat{\varphi}_3$
Old position

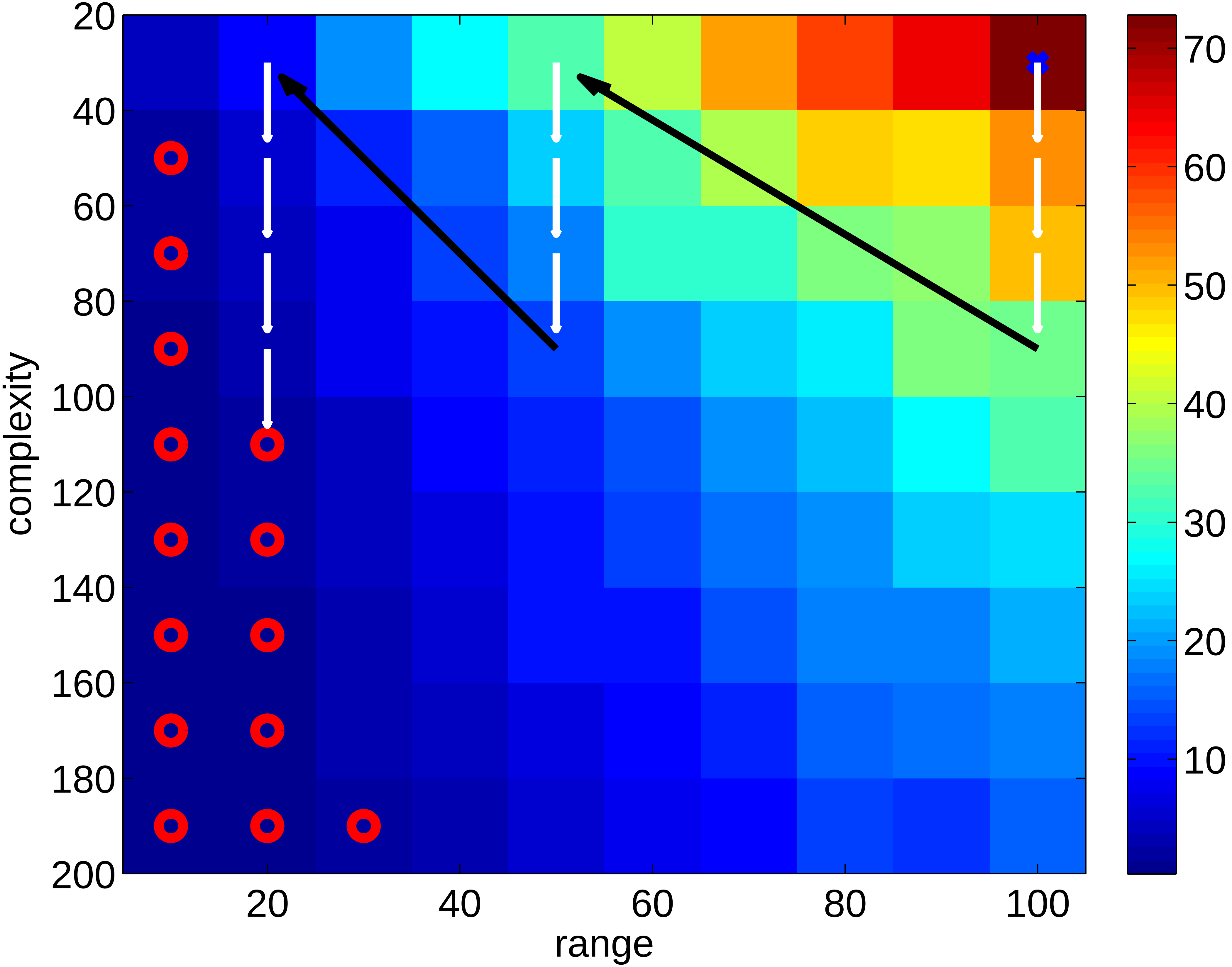

Range
Uncertainty
region
Complexity



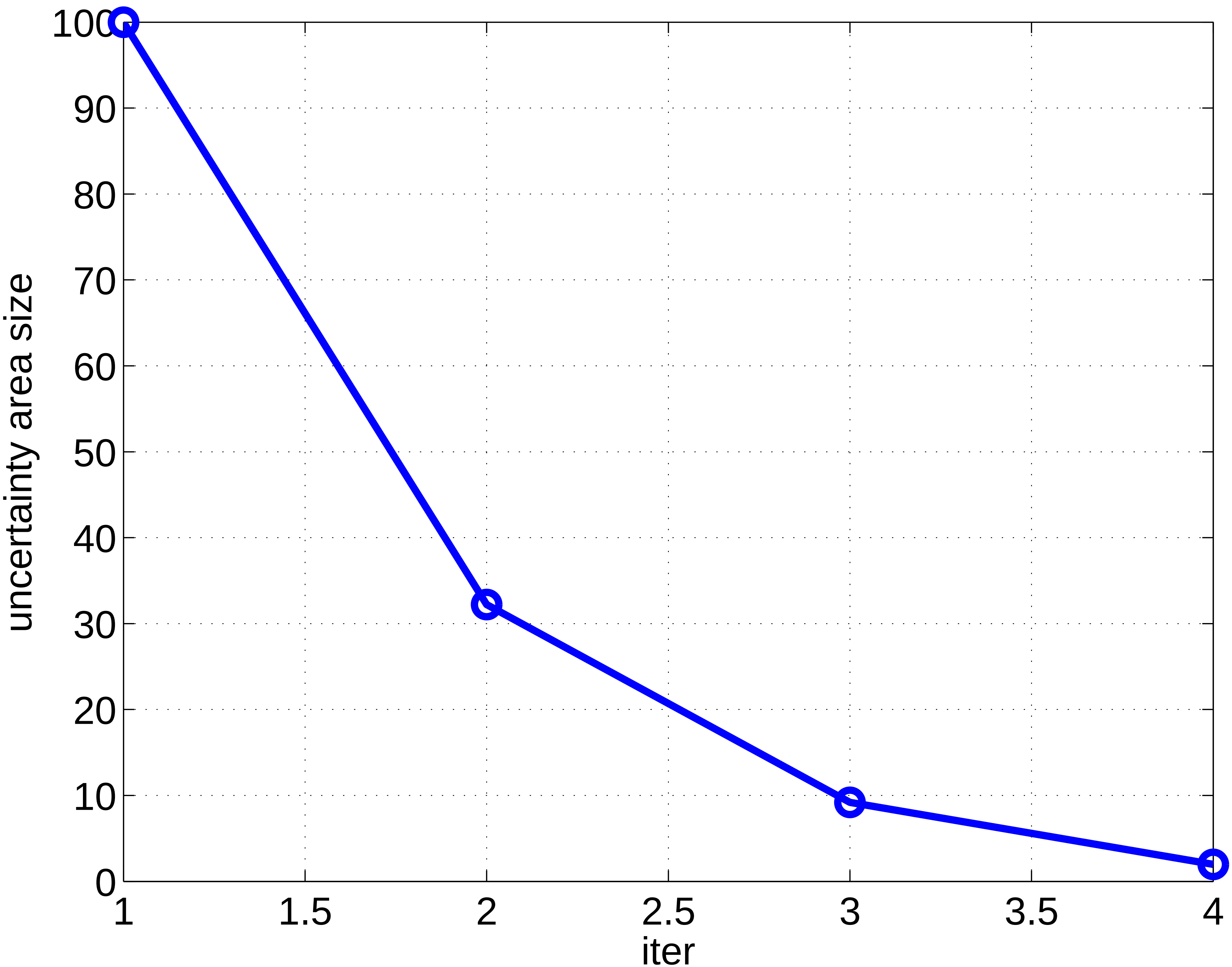
λ−lower bound
Achievable
estimators ω
Inachievable
estimators
λ − uncertainty region
50
40
30
20
10
0
0
50
100
150
200
c − complexity
0
20
40
60
80
100
r − range


λ−lower bound
Achievable
estimators ω
Inachievable
estimators
λ − uncertainty region
50
40
30
20
10
0
0
50
100
150
200
c − complexity
0
20
40
60
80
100
r − range

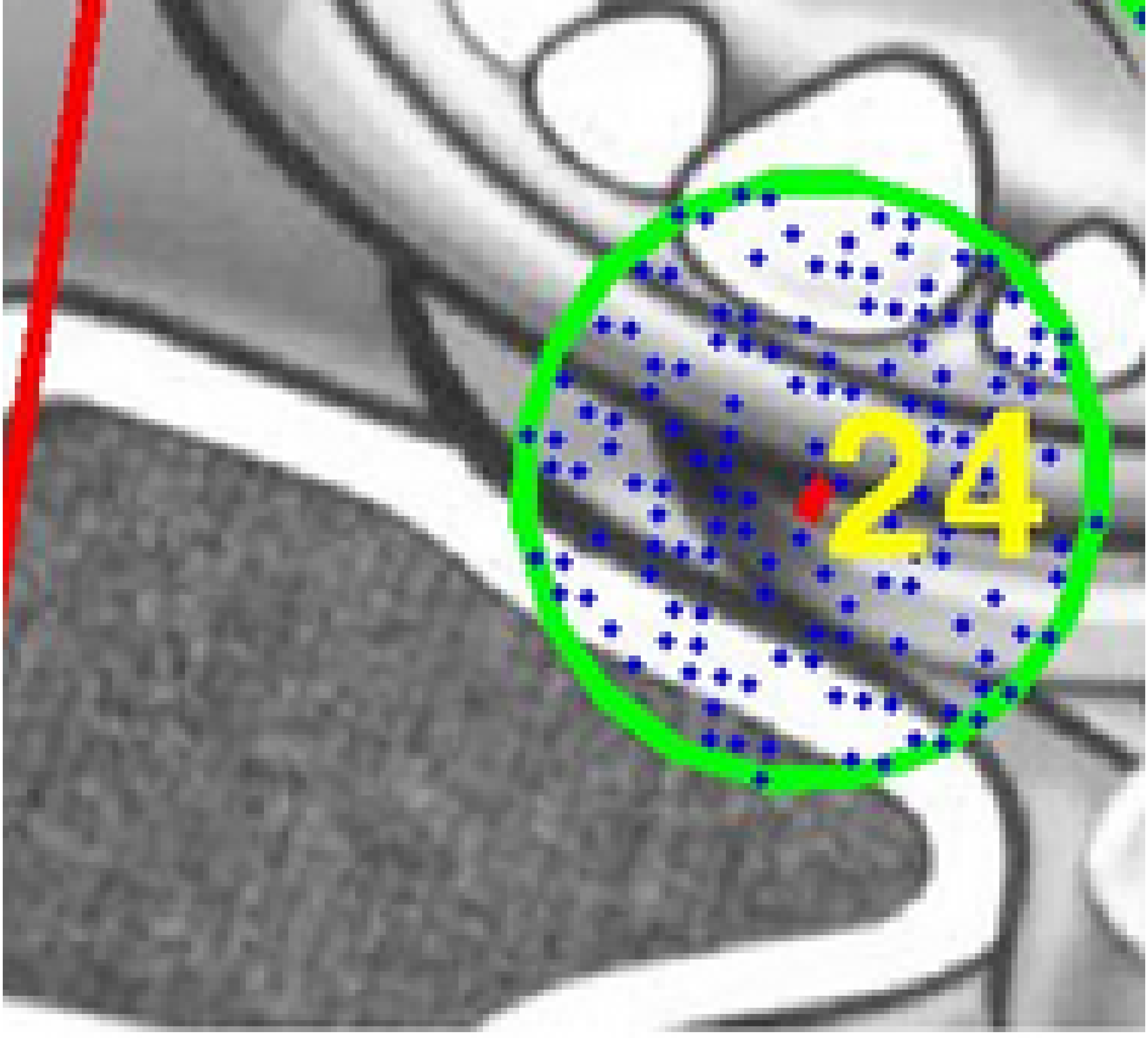

20
40
60
80
100
120
140
160
180
200
complexity
10
20
30
40
50
60
70
80
90
100
range

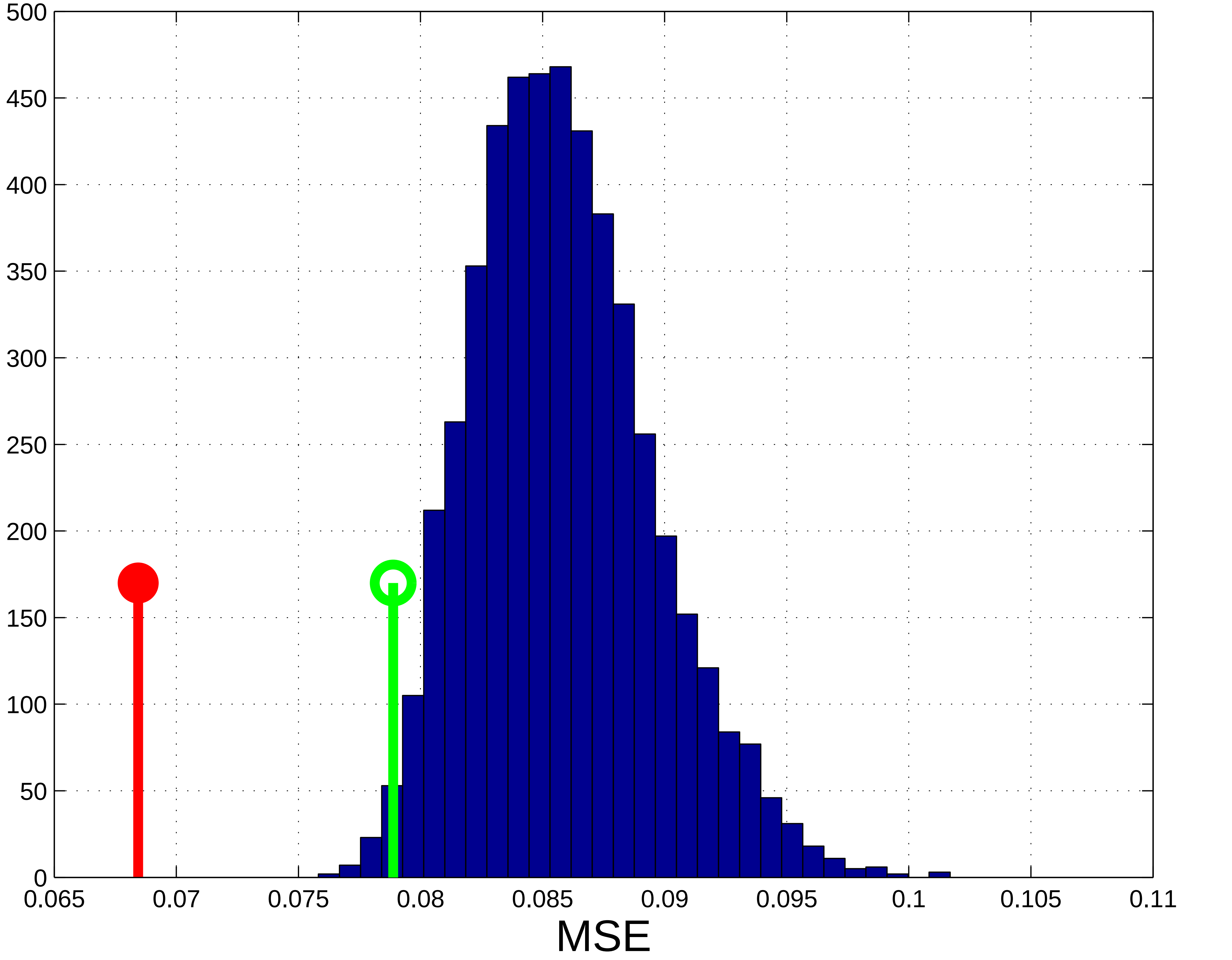


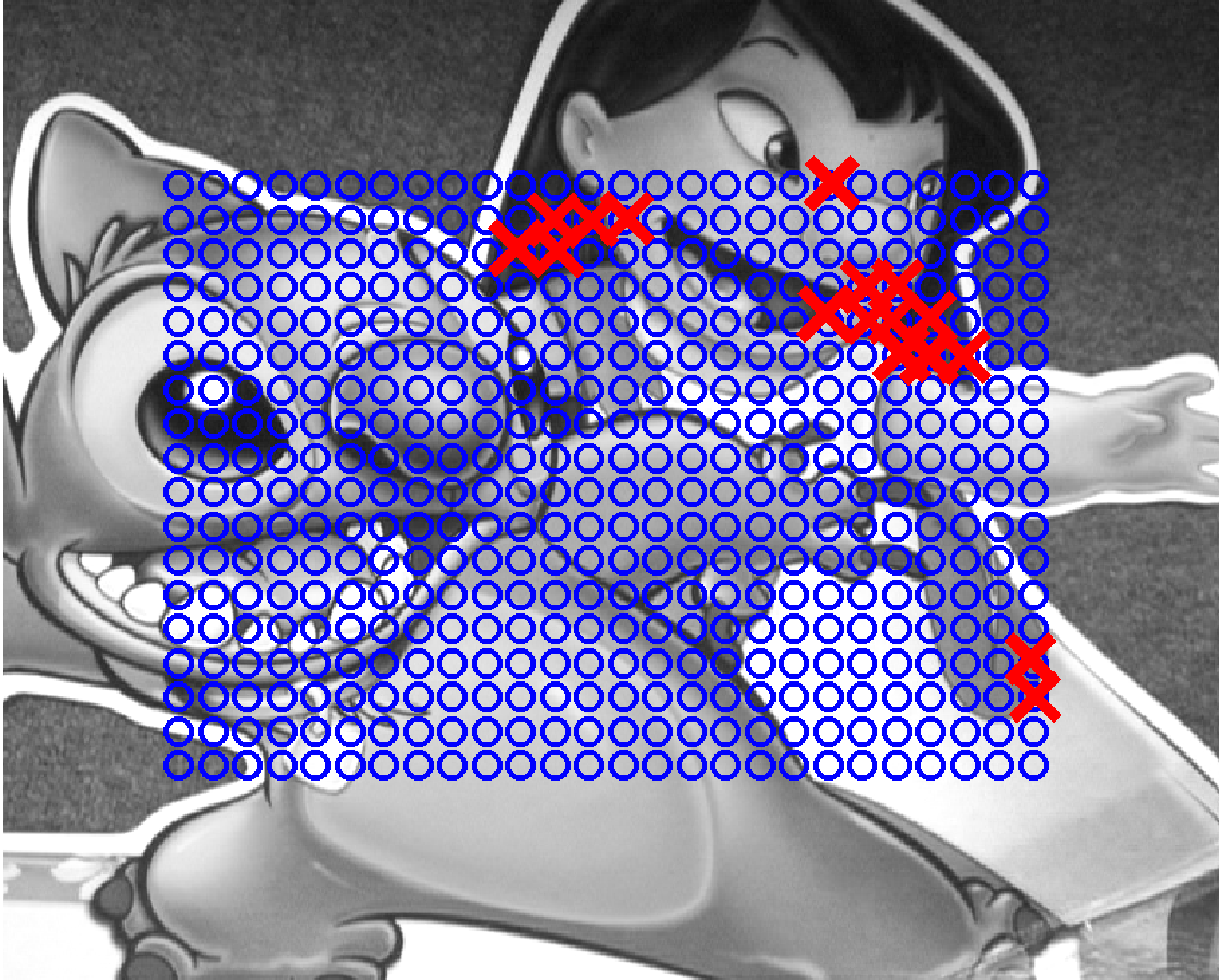
complexity
range
20
40
60
80
100
120
140
160
180
200
20
40
60
80
100
70
60
50
40
30
20
10

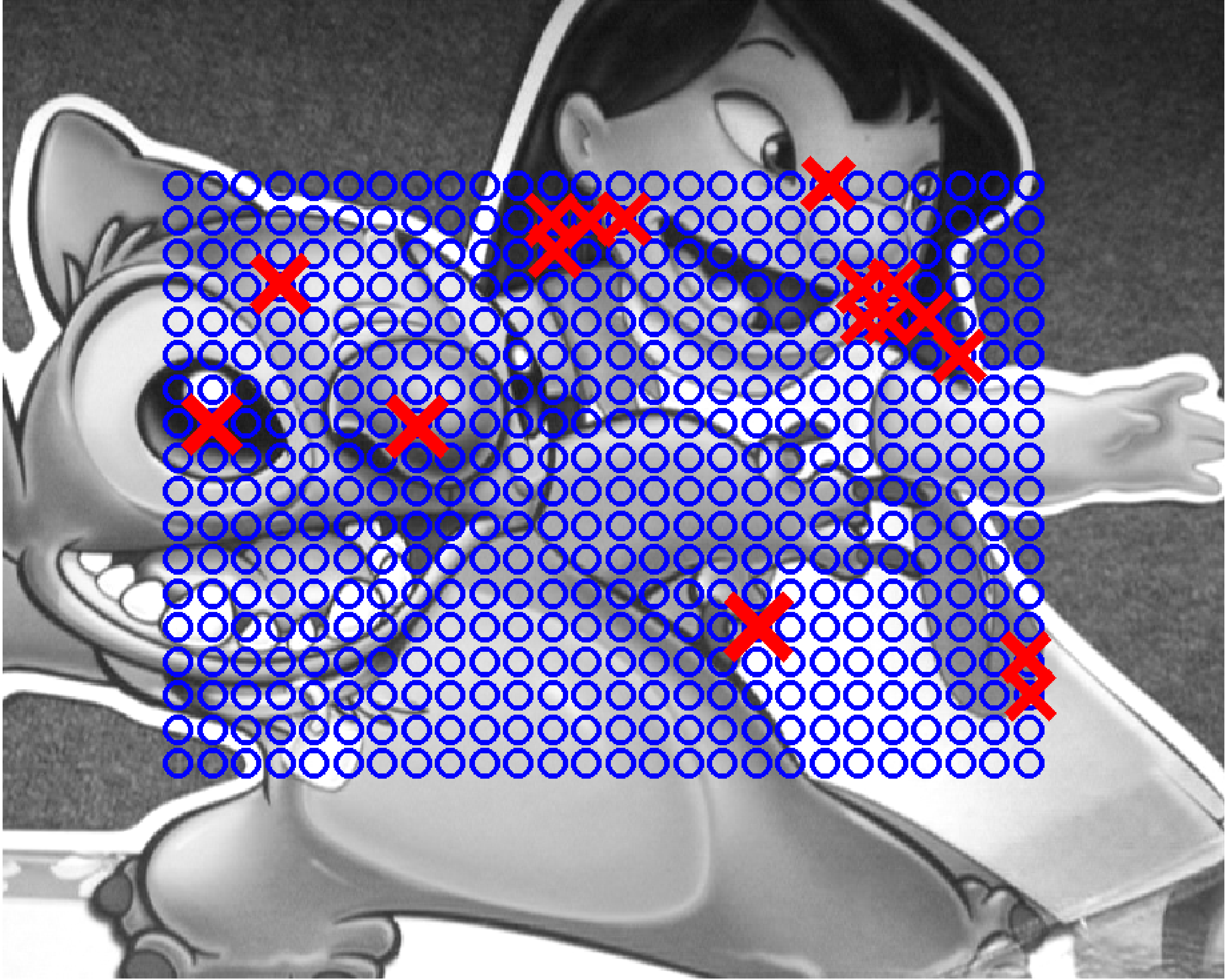


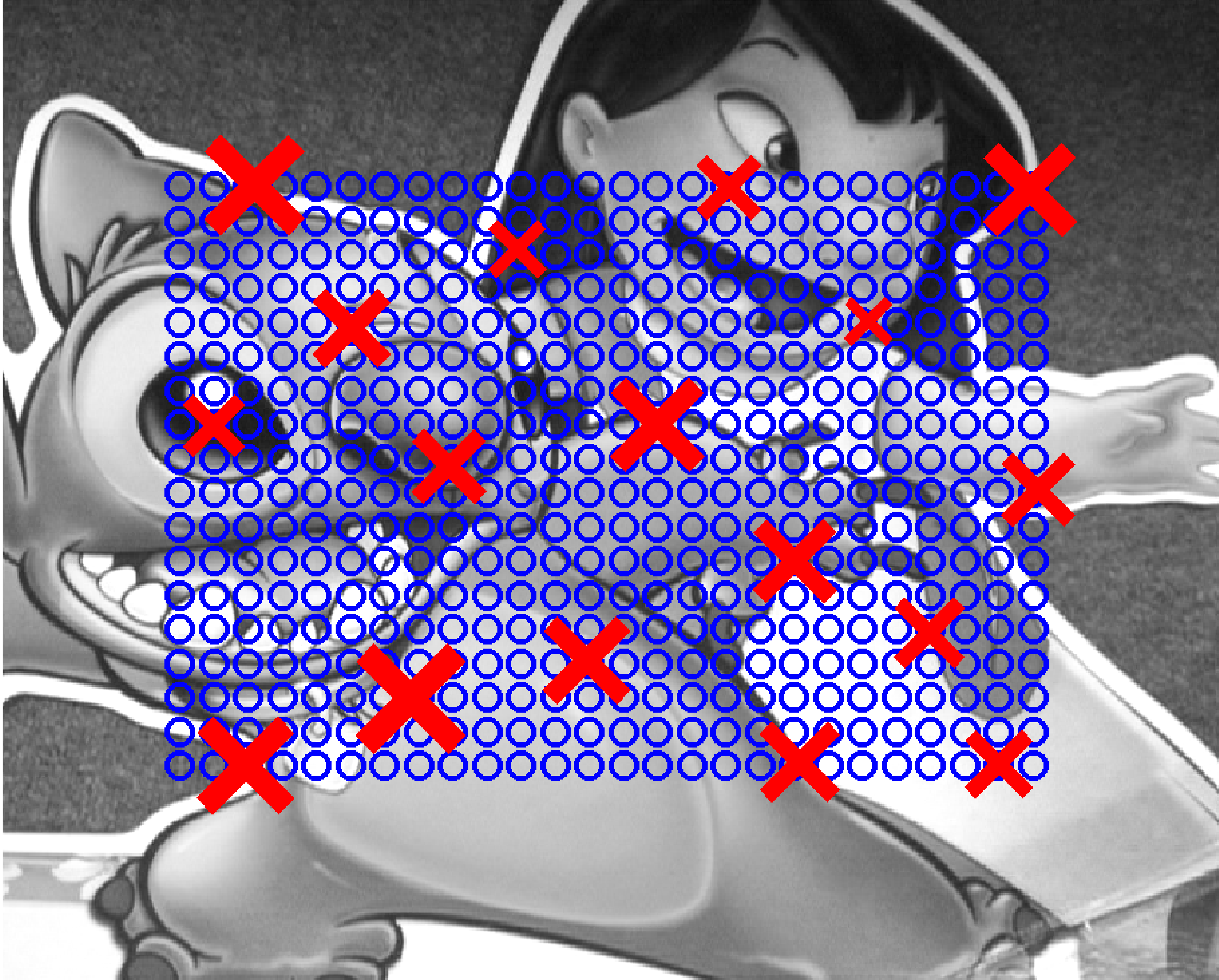
uncertainty area size
100
90
80
70
60
50
40
30
20
10
0
1
1.5
2
2.5
3
3.5
4
iter

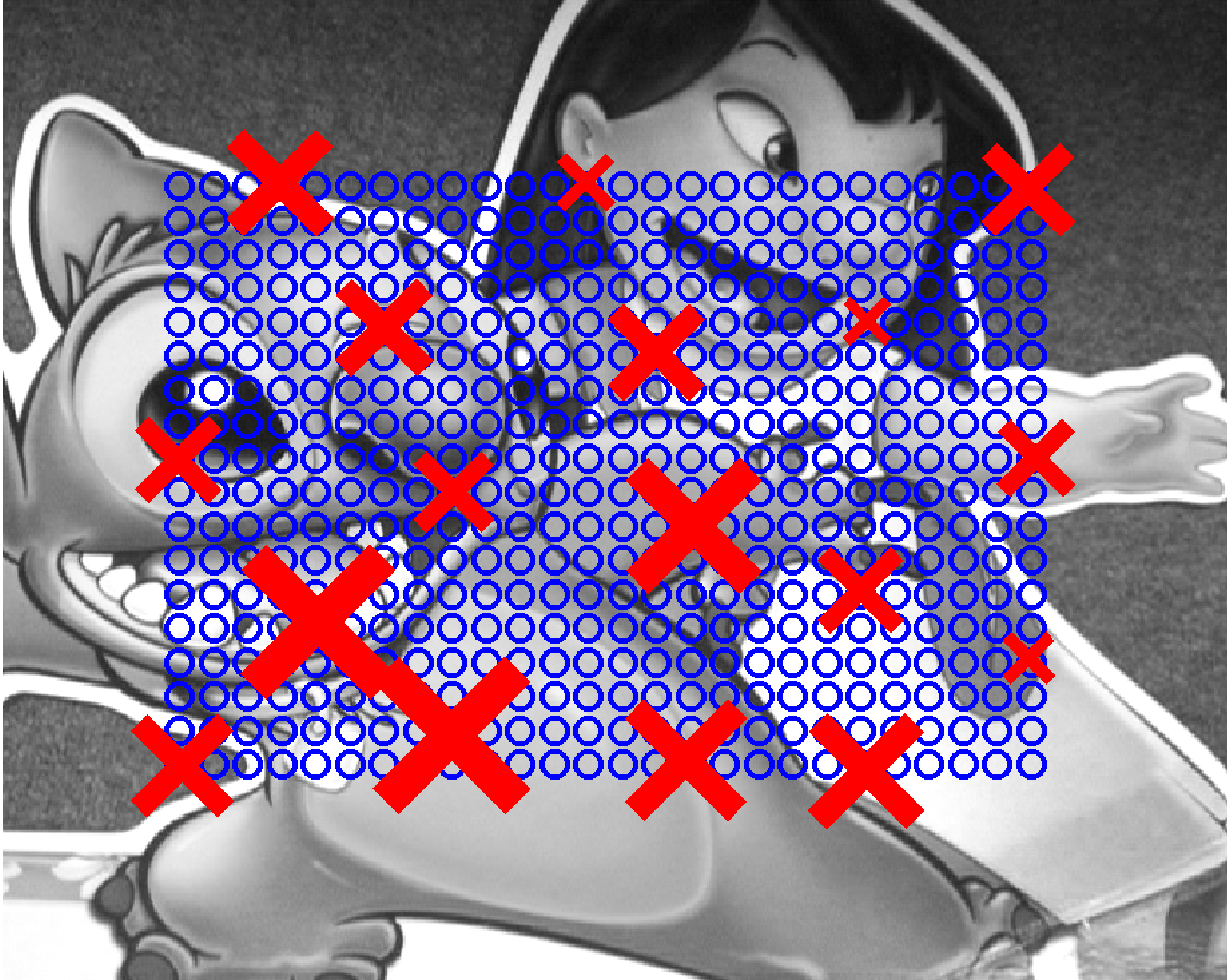

24

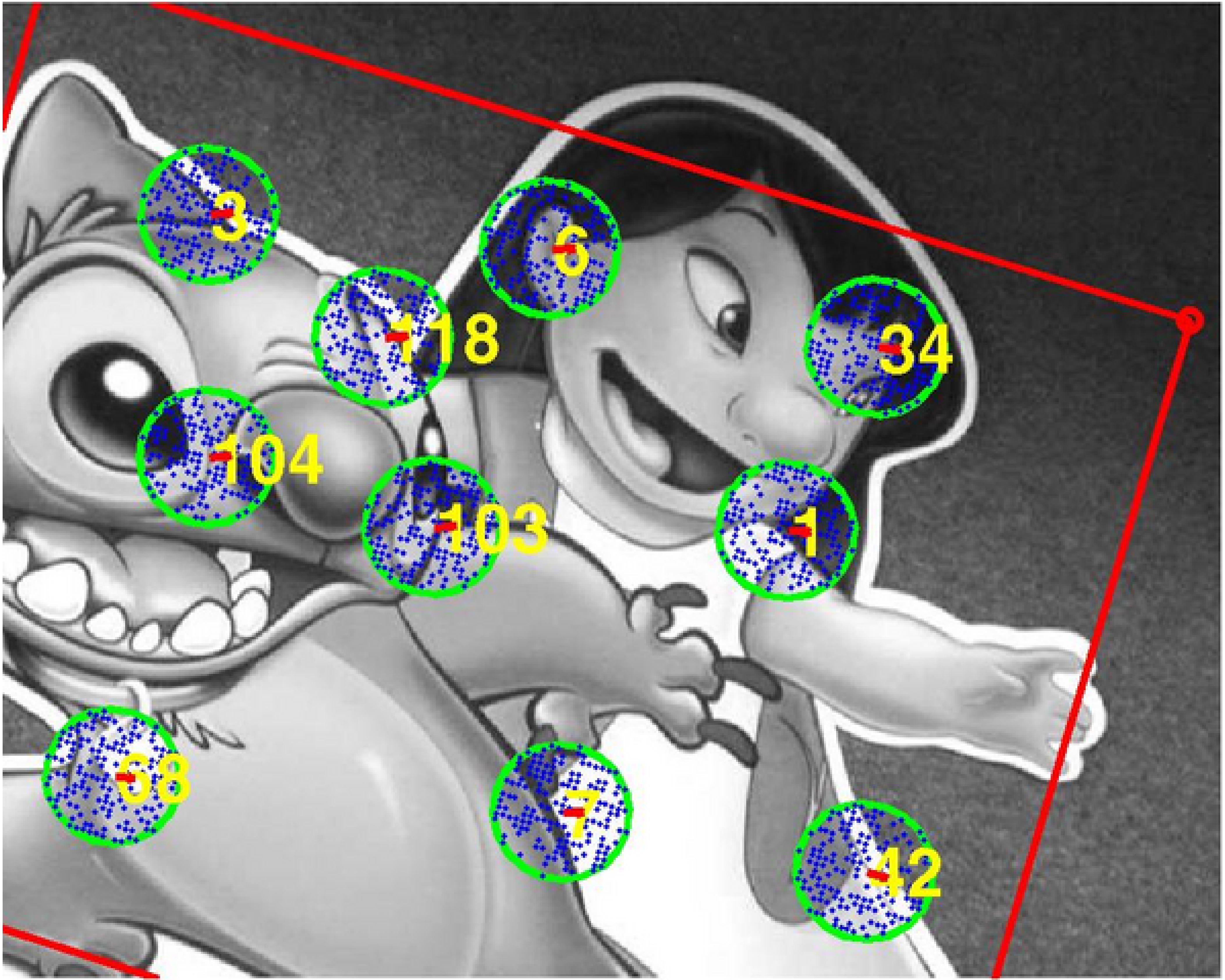

500
450
400
350
300
250
200
150
100
50
0
0.065
0.07
0.075
0.08
0.085
0.09
0.095
0.1
0.105
0.11
MSE



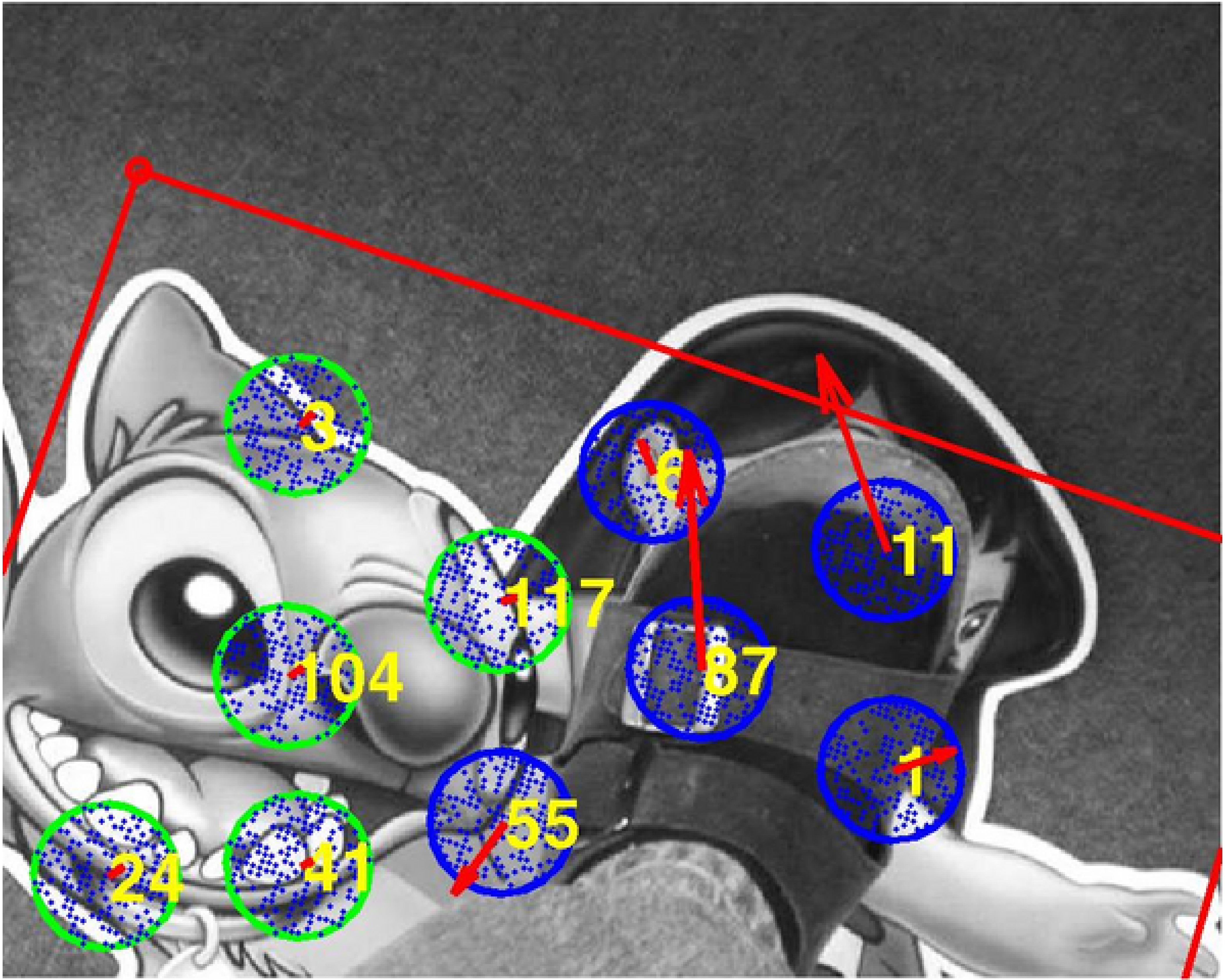


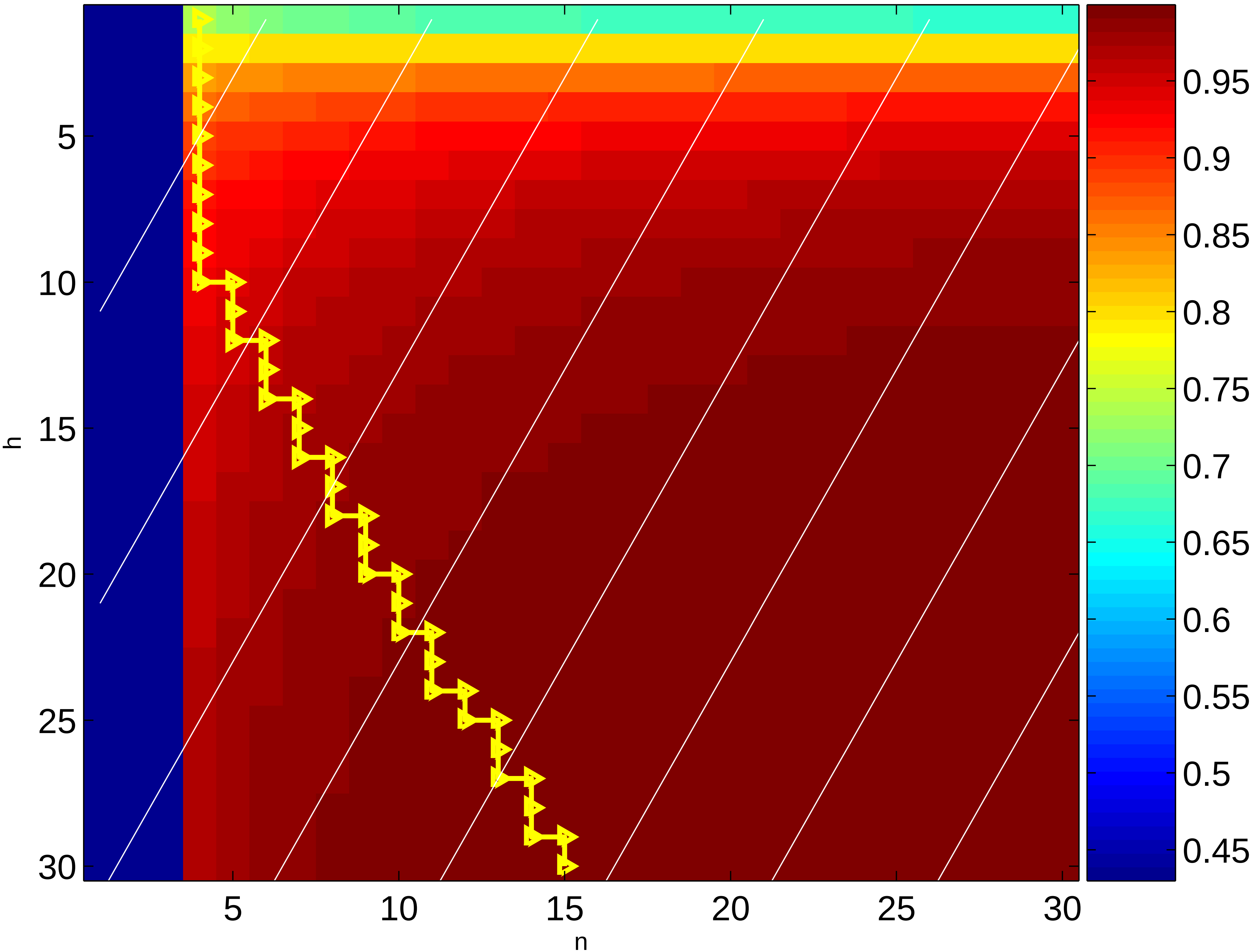




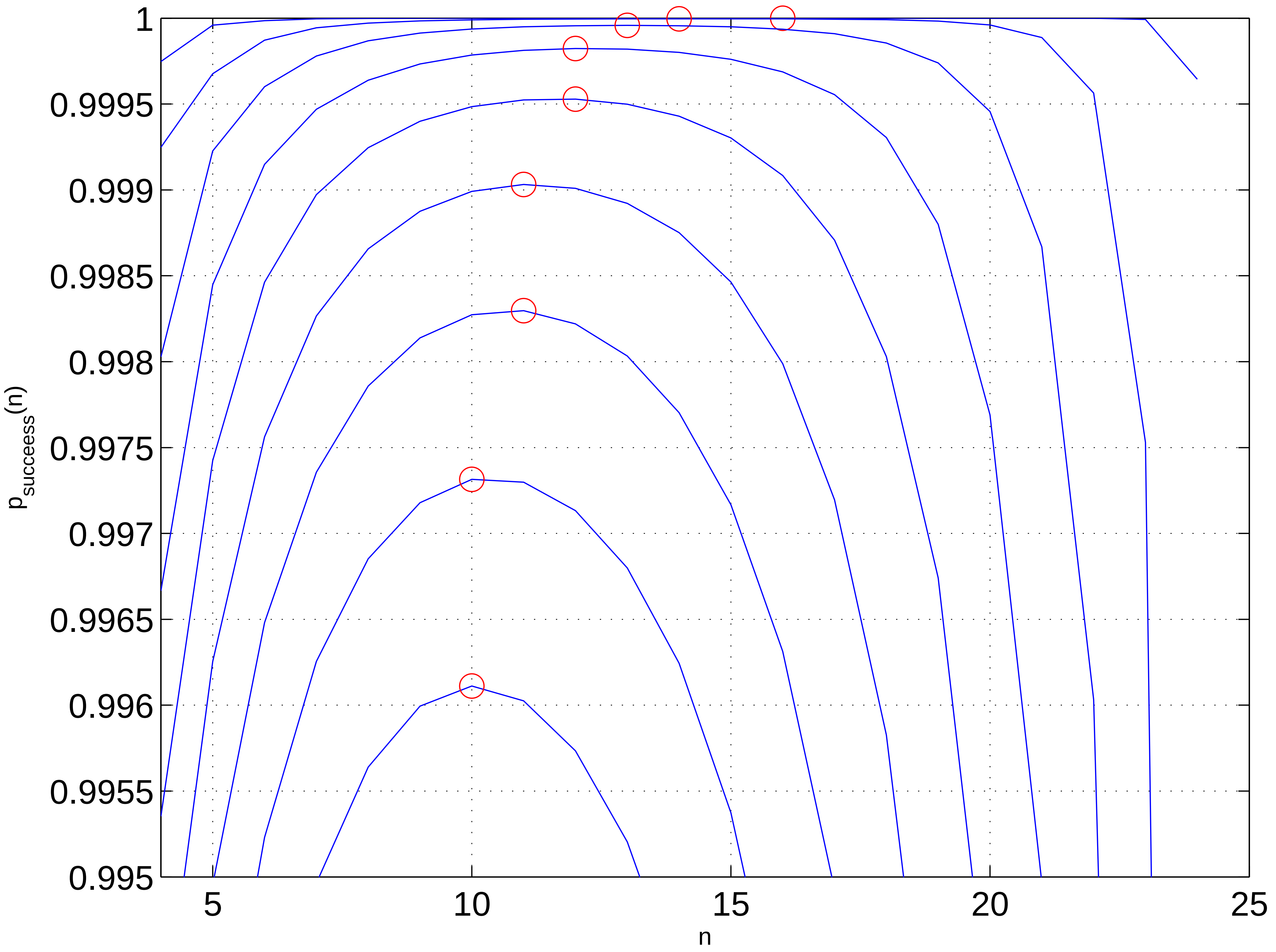


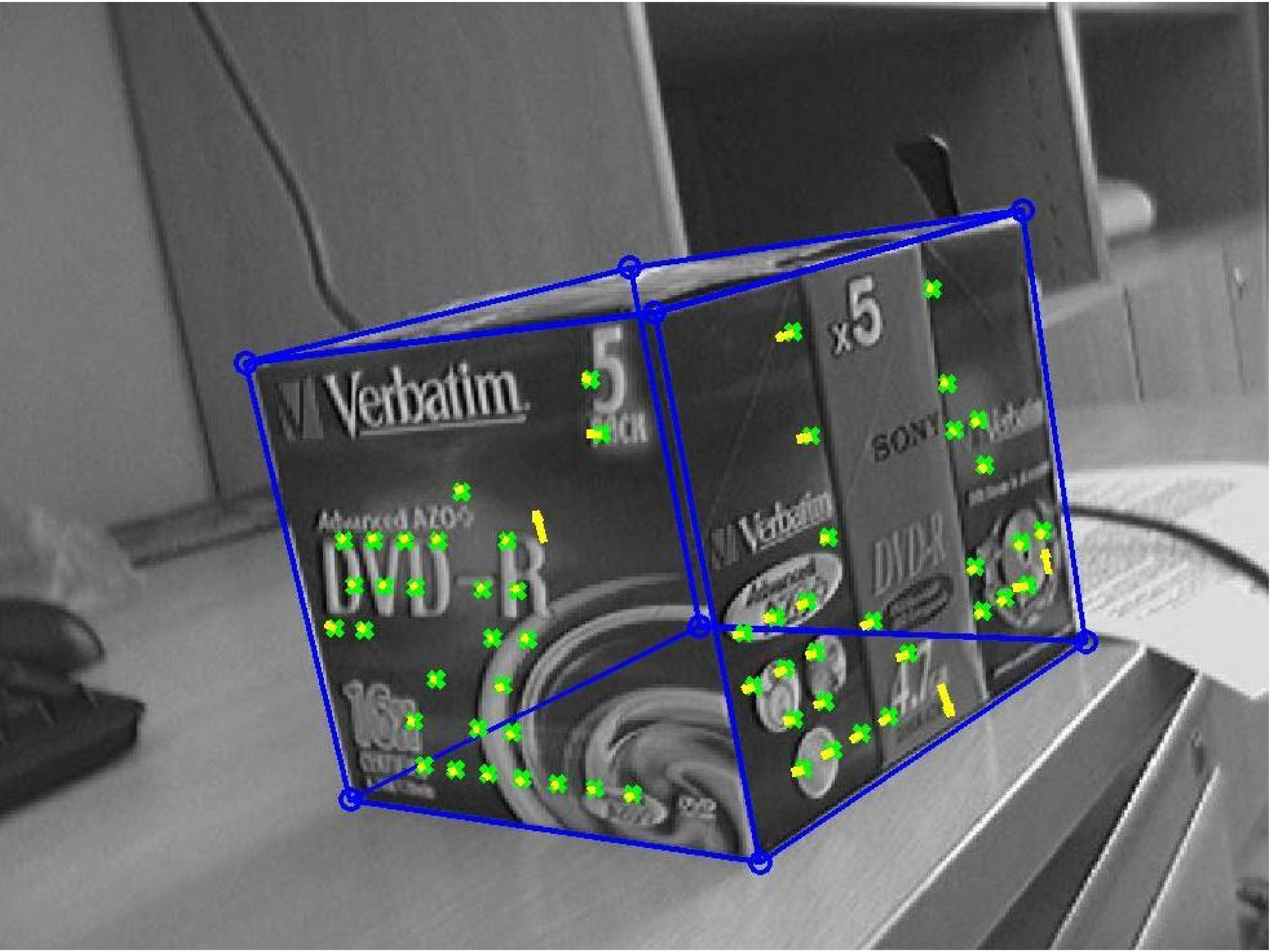
3
6
118
34
104
103
1
68
7
42

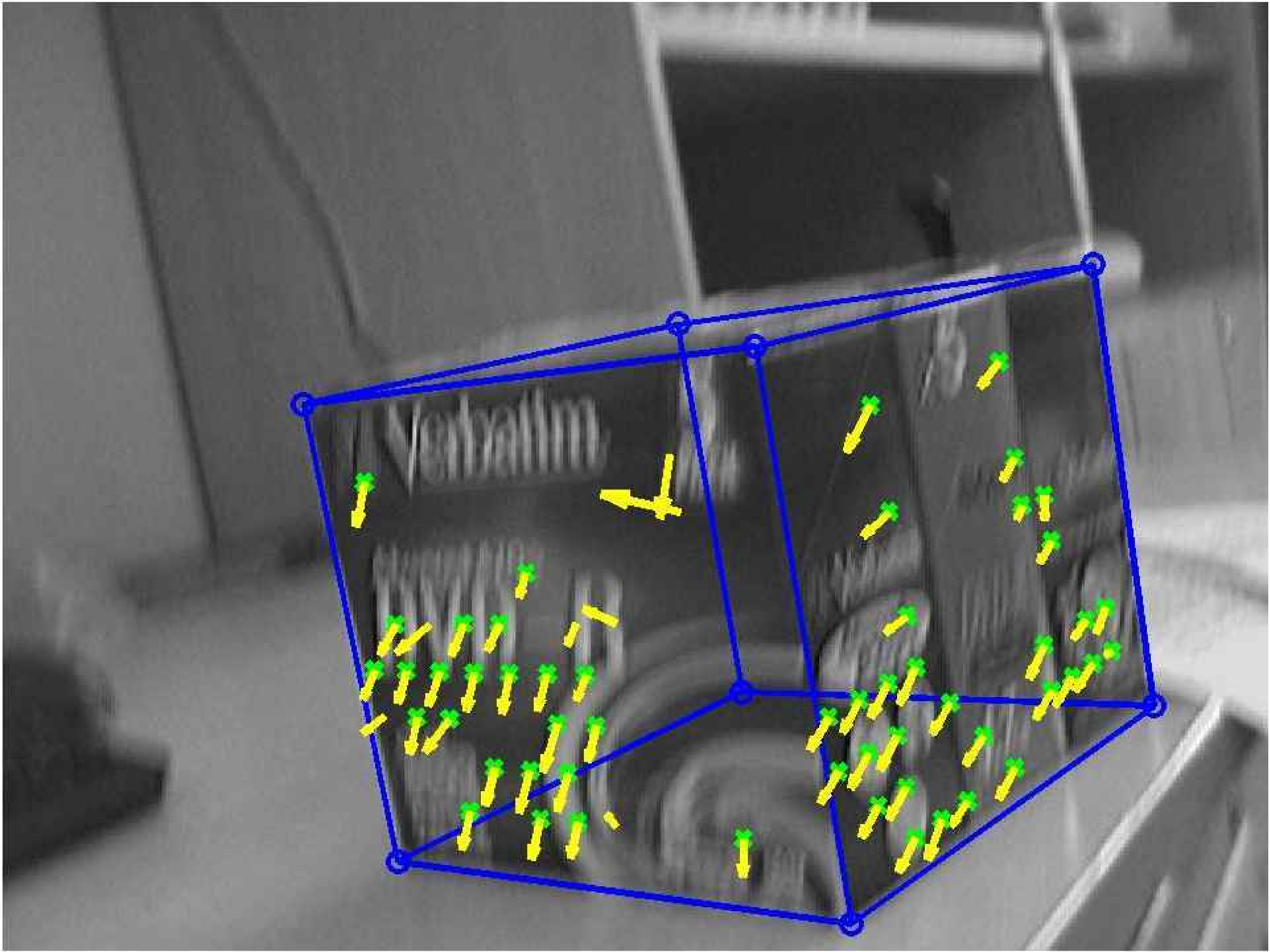

3
6
11
117
104
87
1
55
24
41

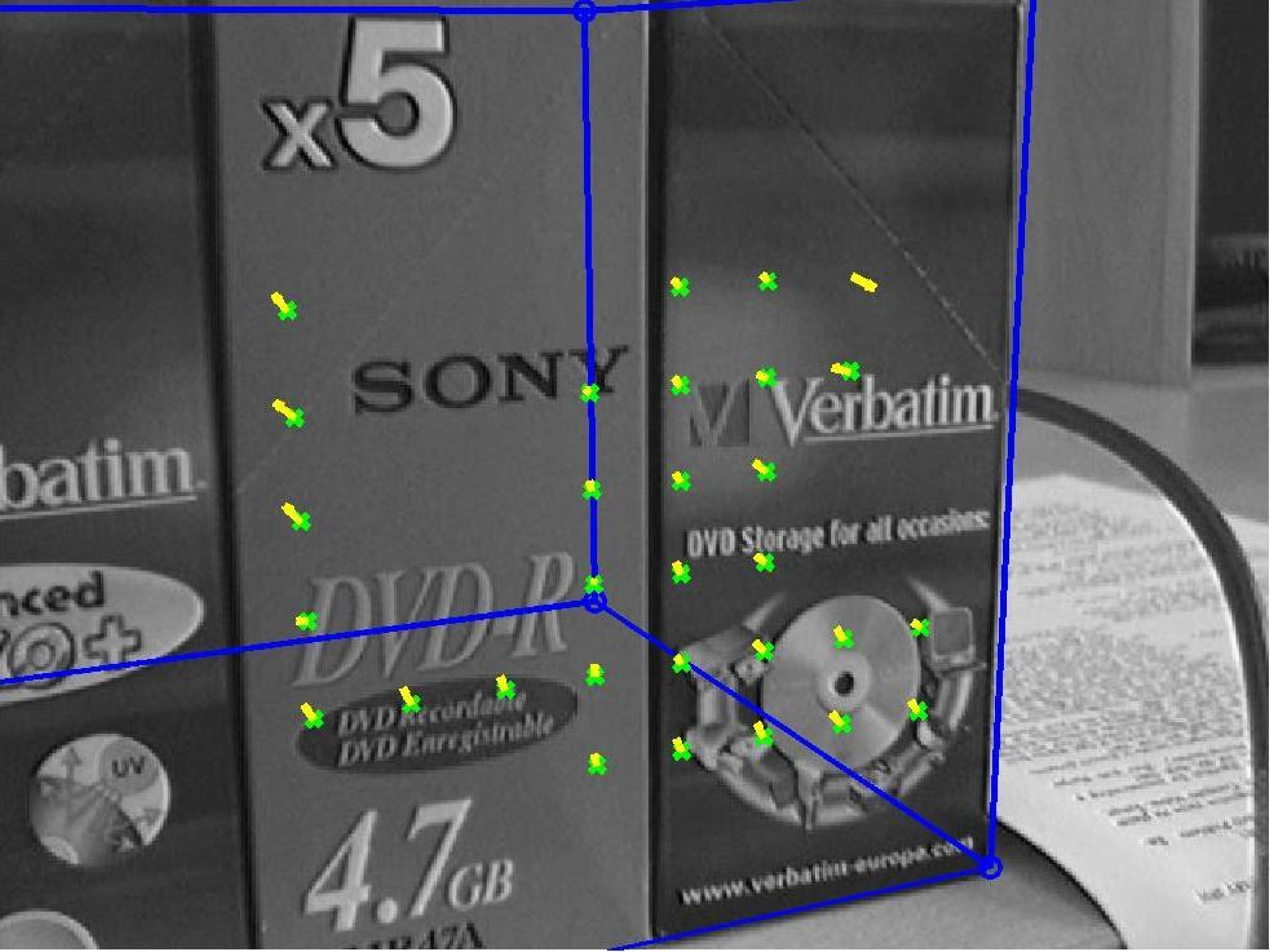


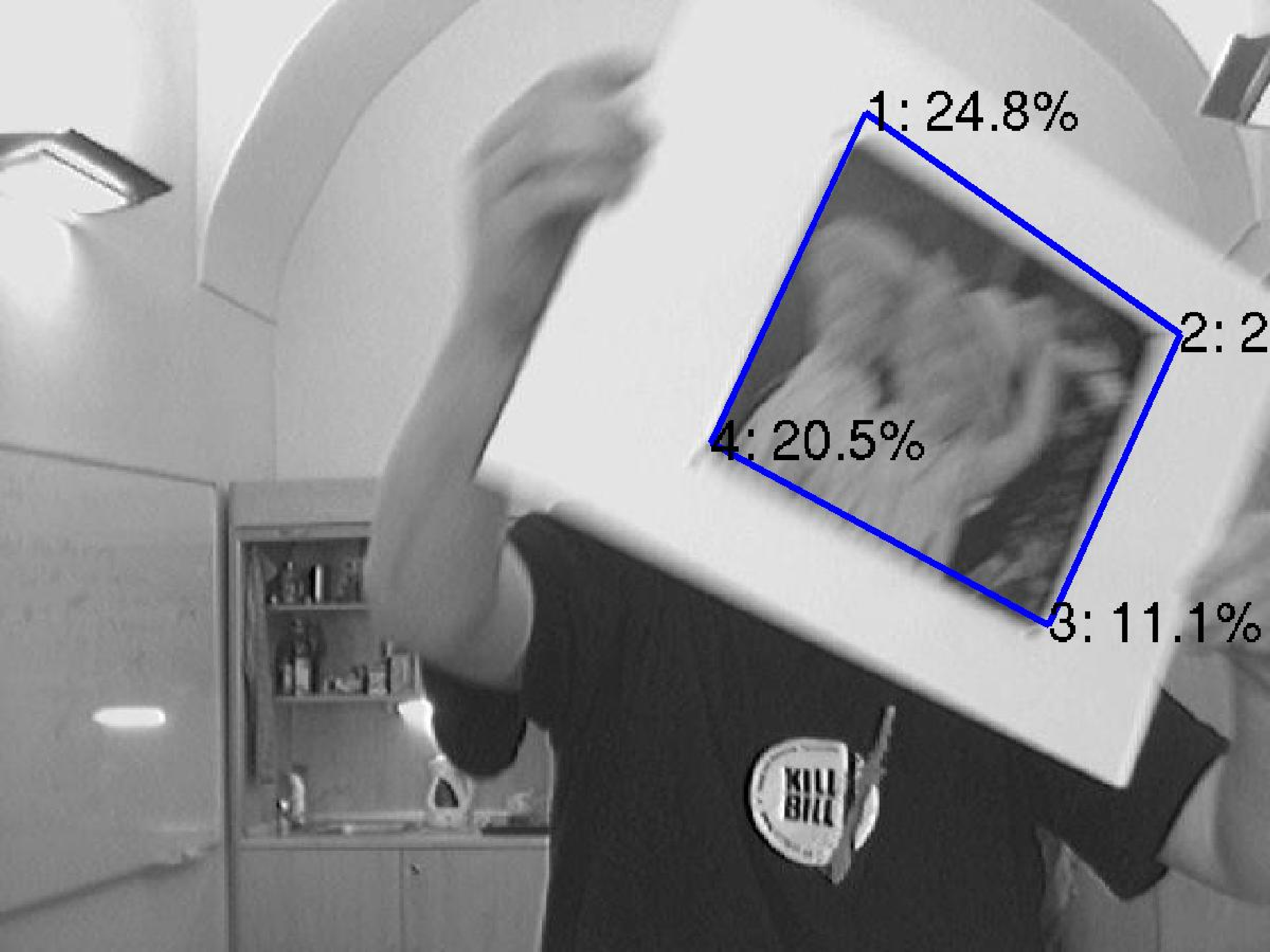
5
10
15
20
25
30
h
5
10
15
20
25
30
n
0.95
0.9
0.85
0.8
0.75
0.7
0.65
0.6
0.55
0.5
0.45

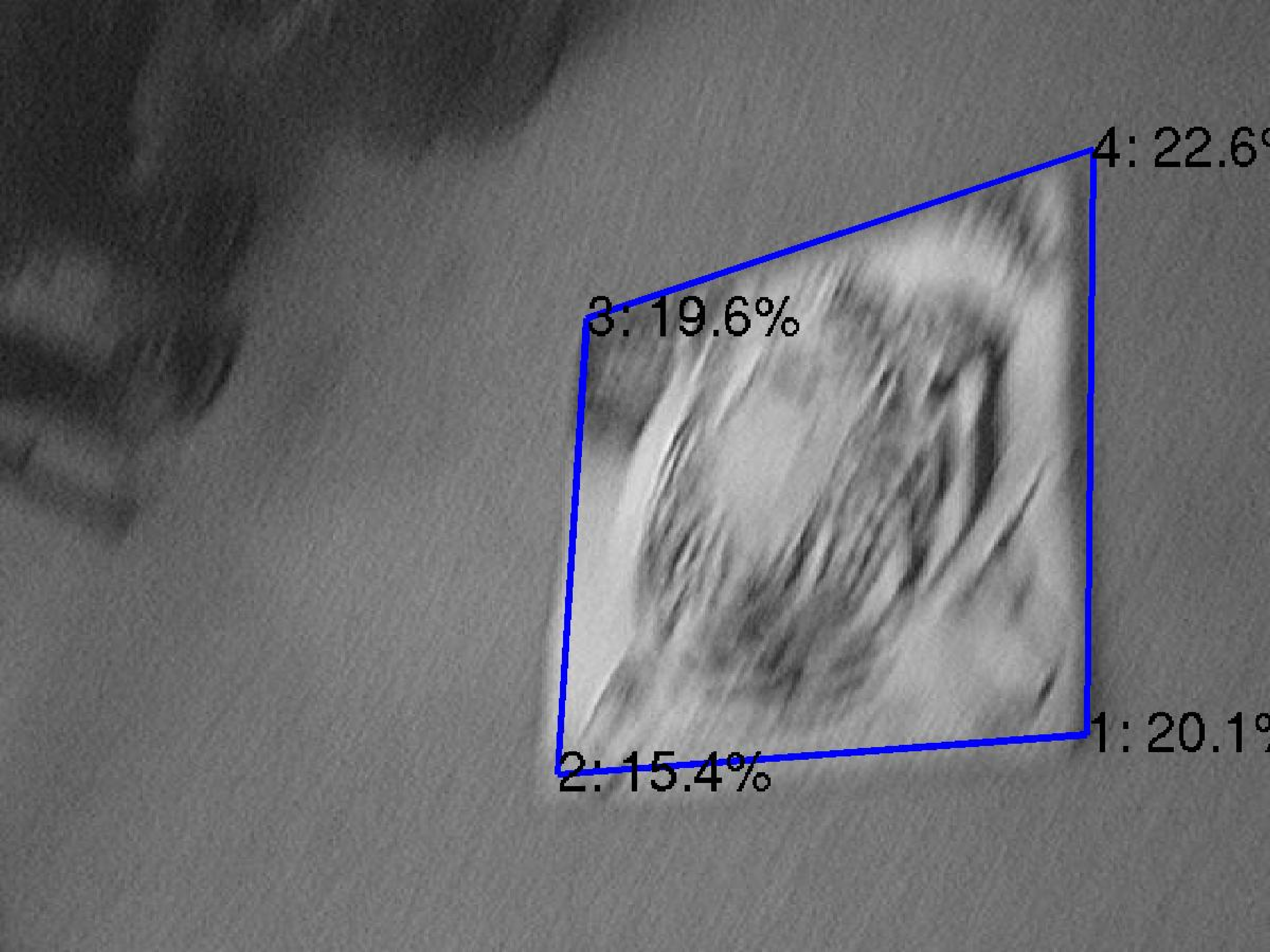

1
0.9995
0.999
0.9985
0.998
0.9975
0.997
0.9965
0.996
0.9955
0.995
5
10
15
20
25
n
p_succeess(n)

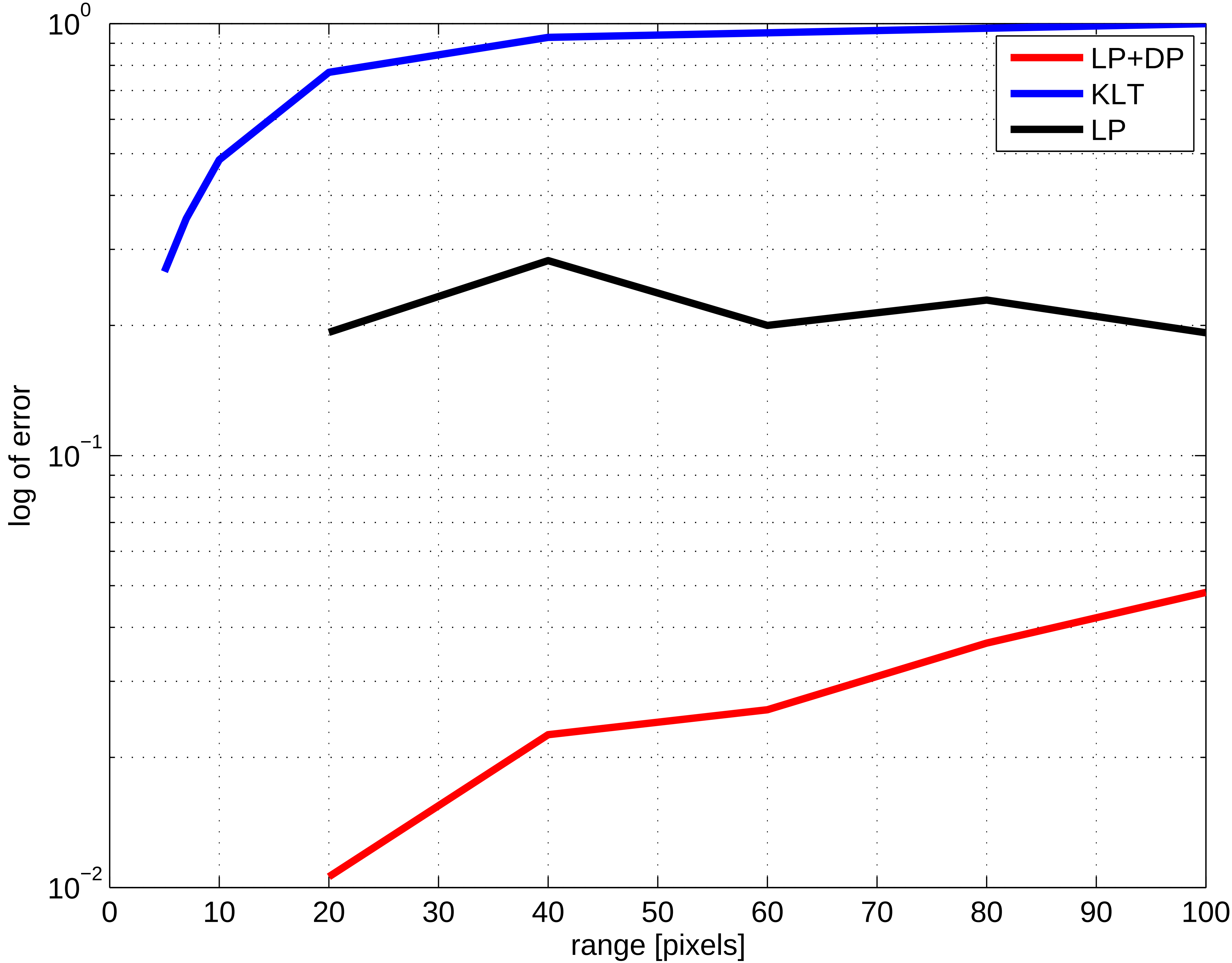

Verbatim
5
DVD-R
x5
SONY
Verbatim
DVD-R

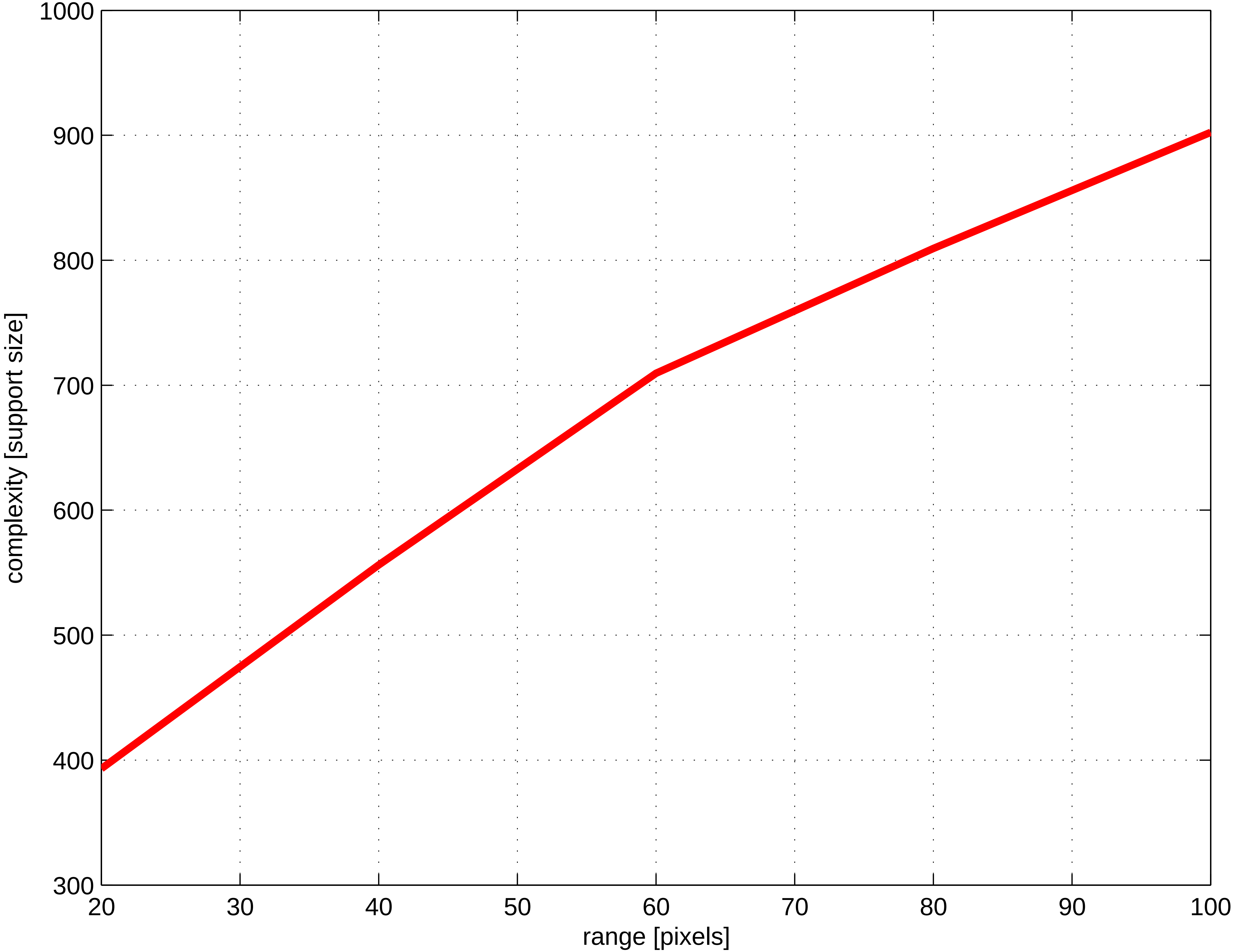


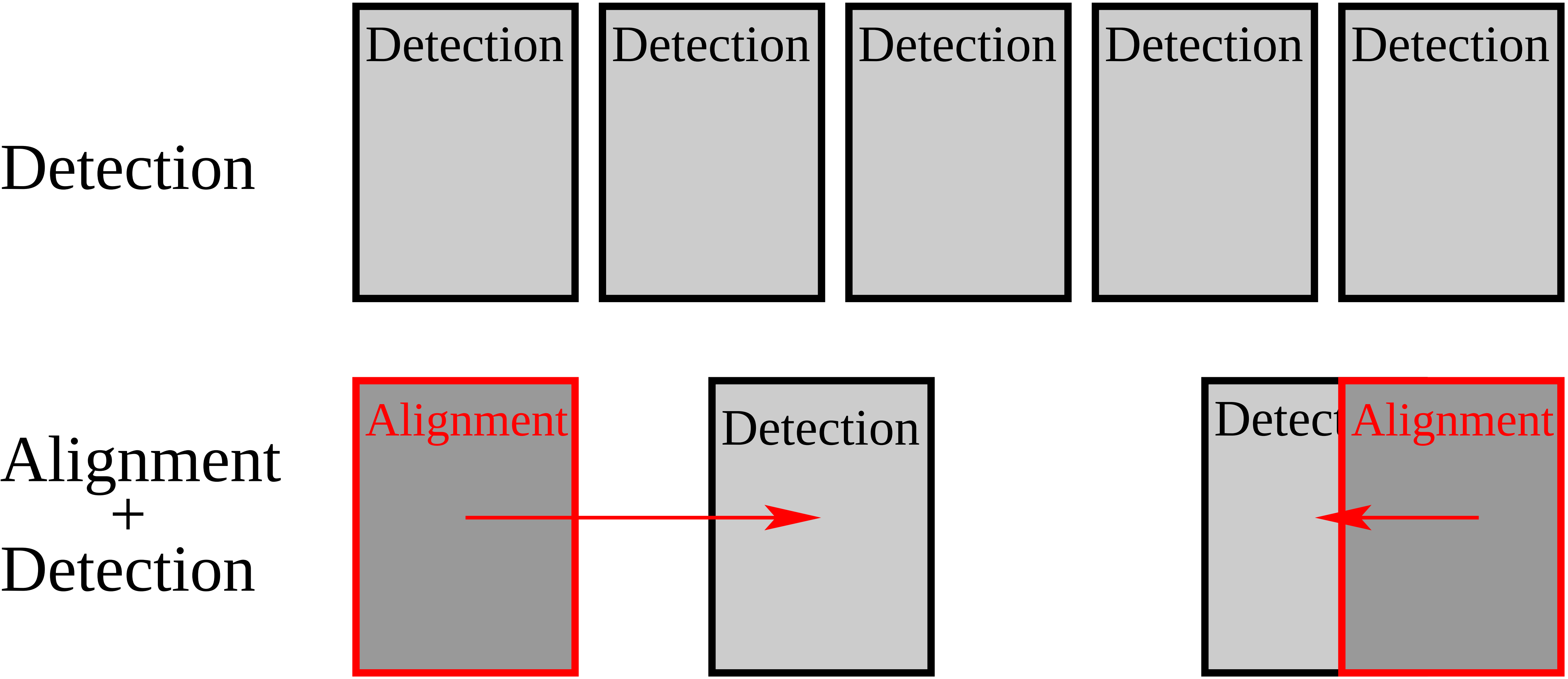

x5
SONY
DVD-R
DVD Recordable
DVD Enregistrable
4.7GB
Verbatim
DVD Storage for all occasions:
www.verbatim-europe.com
batim.
nced
UV

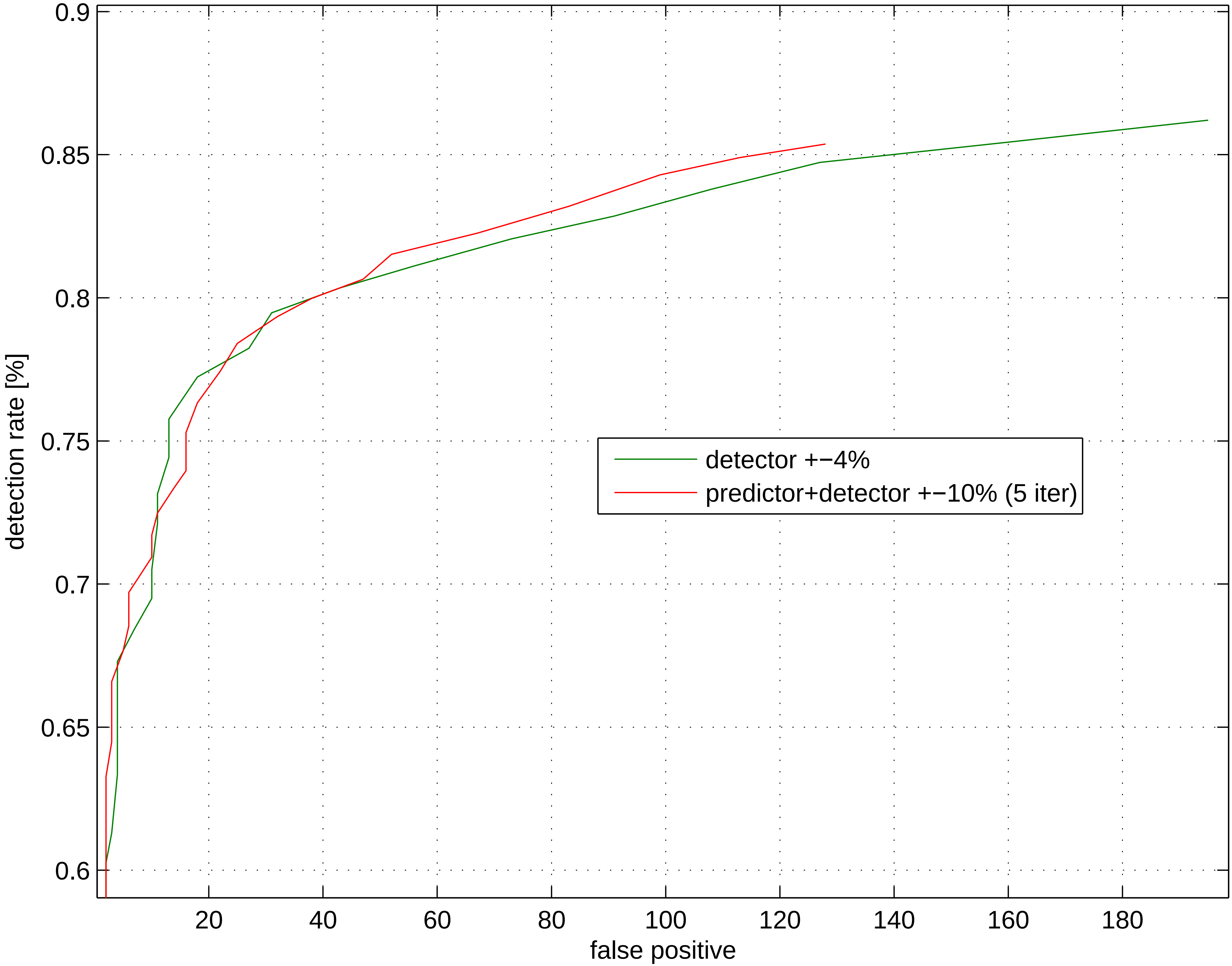

1: 24.8%
2: 2
4: 20.5%
3: 11.1%
KILL BILL


4: 22.6
3: 19.6%
1: 20.1
2: 15.4%


LP+DP
KLT
LP
log of error
range [pixels]
10^0
10^{-1}
10^{-2}
0
10
20
30
40
50
60
70
80
90
100


complexity [support size]
range [pixels]
1000
900
800
700
600
500
400
300
20
30
40
50
60
70
80
90
100


Detection
Detection
Detection
Detection
Detection
Detection
Alignment
+
Detection
Alignment
Detection
Detect
Alignment



0.9
0.85
0.8
0.75
0.7
0.65
0.6
detection rate [%]
20
40
60
80
100
120
140
160
180
false positive
detector +−4%
predictor+detector +−10% (5 iter)